UNIVERSITÀ DEGLI STUDI DI PADOVA
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN
INGEGNERIA INFORMATICA

MASTER THESIS

HEAD RELATED TRANSFER FUNCTION SELECTION TECHNIQUES APPLIED TO MULTIMODAL ENVIRONMENTS FOR SPATIAL COGNITION

Advisor: Ch.mo Prof. Federico Avanzini
Co-Advisor: Ing. Michele Geronazzo

Candidate: Alberto Bedin

ACADEMIC YEAR 2012 - 2013

Senza retorica ma con semplicità
a chi mi ha voluto, mi vuole e mi vorrà bene.

A Giulio, il Telecomunicazionista.
Sei un grande.

Harold Whittles, photographed by Jack Bradley at the very first time he could ear a sound after an earpiece implant.

ABSTRACT

This thesis studies, implements and tests the *pinna reflection model* applied to a user's ear image in order to extract the relevant anthropometric elevation cues for the synthesis of *binaural* spatial audio through headphones. These cues are used for the selection from a database of the best possible *Head Related Transfer Function* (HRTF) for the user. The thesis describes a complete psychoacoustic experiment designed in order to test the performances of the selected HRTF and to compare them with those recorded from the Knowles Electronic Manikin for Acoustic Research (KEMAR) built with average anthropometric data. The pinna reflection model increases the average performances of 10% with a peak of 34% in respect to the KEMAR. It also significantly enhances the externalization and reduces the up/down confusions. The selected HRTF is then integrated in a virtual multimodal environment together with the haptical *TActile MOuse* (TAMO) device, showing the relation and the benefit of audio in tasks that regard spatial cognition, Orientation and Mobility (O&M) and education/training such as goal reaching, reconstruction, displacement and size estimation. The results confirm that spatial audio increases the performance from 10% to 30% in respect to only haptic explorations.

SOMMARIO

Questa tesi studia, implementa e testa il modello delle riflessioni sulla pinna applicato all'immagine dell'orecchio di un soggetto con lo scopo di estrarre le caratteristiche antropometriche rilevanti legate all'elevazione, per la sintesi di audio spazializzato binaurale attraverso cuffie. Queste caratteristiche sono usate per la selezione da un database della miglior possibile *Head Related Transfer Function* (HRTF) per il soggetto. La tesi descrive un completo esperimento psicoacustico realizzato con lo scopo di testare le performance dell'HRTF selezionata e compararle con quelle del Knowles Electronic Manikin for Acoustic Research (KEMAR) costruito con dati antropometrici medi. Il modello delle riflessioni sulla pinna migliora le performance medie del 10% con un picco del 34% rispetto alla KEMAR. Inoltre migliora significativamente l'esternalizzazione e riduce le up/down confusions. L'HRTF selezionata è poi integrata in un ambiente multimodale virtuale insieme al device aptico *TActile MOuse* (TAMO), mostrando la relazione e il beneficio dell'audio nei task di cognizione spaziale, Orientamento e Mobilità (O&M) ed educazione/training come il raggiungimento di un goal, la ricostruzione, il posizionamento e la stima delle dimensioni. I risultati confermano che l'audio spazializzato migliora le performance dal 10% al 30% rispetto alla sola esplorazione aptica.

CONTENTS

LIST OF FIGURES

1 PREFACE

Technology evolution and definitions

Following the incredible quick evolution of computational power, it is becoming easier and easier to create low cost real time multimodal environments. In this work we take advantage of headphones *binaural* audio and *haptic* interfaces to tweak *multimodal environments*. Binaural audio consists in providing binaural (related to two ears) hearing to the user. Binaural hearing, exploiting frequency cues, allows a user to determine the sound origin direction of auditory stimulus presented to both ears. Haptic refers to the touching sensation given by forces imposed over the skin. These forces are conveyed to the brain as kinesthetic information. The term multimodal has been used in many contexts and across several disciplines: in this work a multimodal environment can be considered simply one that responds to inputs in more than one modality or communication channel (e.g. audio and gesture). In spite of its importance, the problem of fusing multiple modalities is sadly often ignored [29].

Thesis goals

This thesis firstly sets out to evaluate the performances of spatial audio rendered with *Head Related Transfer Function*s (*HRTFs*) that have been selected through the analysis of pinna images and to compare these performances with those of generic *HRTFs*. Secondly, this work presents an experimental setup to test a multimodal environment that is just a first step in the creation of cognitive maps for blind people based on spatial audio and haptic devices. The experiment tries to focus on common quality criteria such as "ease of use", "usefulness", learnability, speed and accuracy of user task performance, user error rate and subjective user satisfaction [26] [54].

Thesis overview

Chapter 2 gives a brief overview of 3D audio and multimodal applications in modern societies on different fields: medicine, industry, education, entertainment, arts and blind-aid. It also provides a couple of concrete examples of applications. Then it introduces psychological and math background to both spatial audio rendering and haptic perceptions, focusing on the tools used in this work and proprieties exploited. It also provides an overview of similar previous works. Chapter 3 provides a description of the realized experiments that are the core of this work and that we created in order to evaluate audio performances and multimodal environments. Chapter 4 focuses on the implementation and technical aspects of the experiments, discussing the choices about implementation and showing the main algorithms developed. Chapter 5 presents the result of our work discussing them and highlighting the key points. Chapter 6 shows the conclusions of our works and traces possible future developments. Final appendixes present multimodal maps and extended extracts of the source code used in the experiments.

2 BACKGROUND

This chapter shows the fundamental knowledge required to understand our work. Section 2.1 briefly shows possible 3D Audio and Multimodal Application in our society. Section 2.2 introduces spherical coordinate systems. Section 2.3 introduces spatial audio, its clues and simulation approaches. It also presents the details of the theoretical framework developed by the *Sound and Music Computing Group*[1] at the basis of this thesis. Section 2.4 introduces the haptic device *TActile MOuse (TAMO)* used in our experiments and developed by the *Istituto Italiano di Tecnologia* in Genua as a haptic device to help blind people.

Chapter overview

2.1 3D AUDIO AND MULTIMODAL APPLICATIONS

Applications overview

Multimodal systems and 3D audio have a huge amount of potential and current applications on different fields [6].

In the *industrial* field they can help architects and designers to easily manages 3D projects with software that are nowadays an industry-standard such as the *CAD* family, *3D Studio Max* and *Maya*: the possibility to efficiently navigate in these environments is a key features on these working areas. Another industry application resides in testing the manual assembly process of mechanic components before they are physically realized [47].

In the *education* field they can help kids at different instruction levels, allowing them to touch and sense shapes in order to understand the scale of real objects and in order to give them a concrete and captivating feedback. Kids can also understand macro scales such as astronomical scales and nano environments exploring virtual worlds in a interactive way.

In the *entertainment* field they can be used on video games and simulators; avatars; augmented reality and cinema. Nowadays haptic interfaces for playing with at least 2 degrees of freedom are massively produced and they are relatively cheap. They can simulate particular circumstances e.g. the device can vibrate when the player is above a bridge in order to simulate the driving above planks or the device can provide a sort of resistance to match a real physical conditions or also headphones can reproduce a sound coming from a particular area.

In the *arts* field they can allow the user to go inside a captivating environment and they can help the artist to experiment new form of art providing him or her tools to remotely play musical instruments, virtual sculpting and so on.

In the *medical* field they can be used as hospital staff training tools with surgical simulators; as remote-assistant interfaces using micro and macro robots; as telemedicine systems and as aids for disables.

1 Department of Information Engineering, University of Padua

Last but not least, they can also aid *blind peopl*e on the cognition and exploration of spatial unknown maps, allowing them to easily identify obstacles and positions. Blind people can also consider to download maps before going to a public place in order to already know the configuration of the real place once they will be there. These systems can also aid low-vision people. It has been proved that map recognition and reconstruction is more holistic and comprehensive for people that can explore a virtual environment than those who need to explore a real space. This is because a real space exploration is mainly linear while a virtual environment can be explored freely starting from any possible location and moving in any of all the possible directions with multi-sensory feedback [35].

3D audio provides a valid support to virtual environments creating a more captivating experience. However it has other stand-alone applications such as alarms and warning messages generation, broadcasting and the already cited cinema area. Many undiscovered uses probably exist as well and they could entail either variations on existing system designs or entirely new approaches [4].

Concrete examples

To give just a couple of examples of 3D audio practical applications, we would like to report a desktop *PC browser* that parsed an HTML document structure and it maps it to a 3D audio space. It focuses on key element such as anchors and links and it can improve the effectiveness of the browsing experience performed by blind people [23]. Another quite curious example is the use of 3D audio in a study on the virtual reality *exposure-based treatment for cynophobia*. In these applications, auditory information is not used as a way to supply visual information: the potentiality of 3D audio is exploited in order to increase the realism and the richness of an immersive environment [60].

2.2 COORDINATES SYSTEMS

Coordinates system

Before starting to introduce the background elements of this work, it is useful to present the conventions about coordinates systems adopted. In fact we often need to place virtual spatial points around a person. To define a point position in space, instead of the common Cartesian *x y z* coordinates system, it is a common practice in auditory papers to use a coordinates system that is similar to the polar system. However there are multiple adopted standard so it is critical to clarify our choice. On figure 1 we reported the two main coordinates system. Both are centered with their origin at the center of a subject's head, at his or her ear level. In the first one (*Vertical-Polar Coordinates*) a first angular value called θ(*azimuth*) defines the rotation, in degrees, on the horizontal plane starting from the 0 degree position in the correspondence of the subject's nose. The angular value is *positive* for *clock-wise rotations* and it is *negative* for *counter-clock-wise* rotations. Consequently $\theta \in [-180, +180]$. A second angular value called φ (*elevation*) define the rotation, in degree, on a plane perpendicular to the horizontal plane and intersecting the line that goes from the origin of the system to the

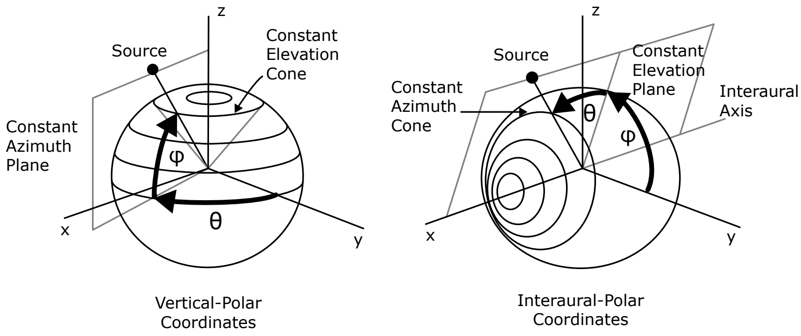


Figure 1: Different coordinates systems. In this work we use the vertical polar coordinates system.

point identify by the azimuth value. A rotation in the *upper* hemisphere has a *positive* φ value, while a rotation in the *lower* hemisphere has *negative* one, so defining $\varphi \in [-90, 90]$ allow to define with (θ, φ) all the points on a sphere centered on the origin. The distance from the center of the sphere is often referred as d or r. It should be noted that there is a similar coordinates system (and it is used by the *Audio Engine* of this work) that defines $\theta \in [0, +360]$ in counter-clock-wise and $\varphi \in [0, 360]$ starting in the upper hemisphere. On the same figure 1 the image on the right (*Interaural-Polar Coordinates*) shows a different system. It works exactly as the previous one, but in order to reach a specific point, position changes are performed firstly on the elevation φ and secondly on the azimuth θ.

In the first system it is possible to identify planes with constant azimuth and cones with constant elevation, while in the second system it is possible to identify planes with constant elevation and cones with constant azimuth. In this work, except where different noted, we use the *Vertical-Polar Coordinates*. The vertical plane with constant azimuth $\theta = 0$ is called *medial plane* or *sagittal plane*.

2.3 BINAURAL PSYCHOACOUSTIC BACKGROUND

2.3.1 Spatial hearing

Spatial sound perception on humans

The human ability to perceive 3D sounds is of great interest and it is only partially understood. It is clear that this ability is partially given by the possibility to ear a sound on both ear and by the presence of the head between them. In this analysis, based on the work of Avanzini *et. al.* [2], we will consider that our brain elaborates the perception of spatial sounds starting only from the two acoustic pressure signals coming to the eardrums. It is not exactly correct because there are other complicated factors that influence the perception of the sound such as the visual stimuli that come to the eyes, the nature of the sound, its timbre, and the propagation of the sound through the bone conduction.

The role of the head

We said that the head has a key role in the perception of 3D sounds.

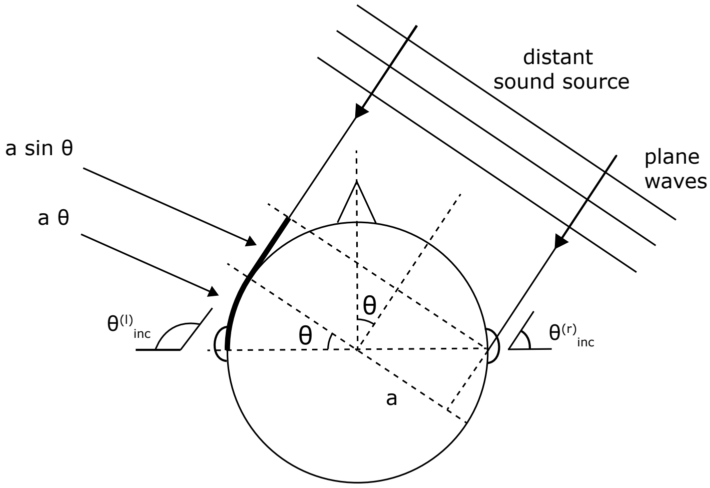


Figure 2: ITD estimation. Assuming that the sound comes as plane waves it reaches the ears at different instances [2].

This is because it is an obstacle to the propagation on the sound, in particular it adds a time delay to the instants in which the sound differently reaches the two ears. These two delays concurrent to cause the *interaural time difference* (*ITD*). ITD can be easily calculated in an approximated (but still useful) way. As shown on figure 2, calling θ the angle that the horizontal median line identifies with plane waves and calling a the radius of the head[2], the ITD is the ratio between the supplementary distance needed to reach the farthest ear and the speed of sound *c:*

$$ITD \approx \frac{d}{c} \approx \frac{a}{c}(\theta + sin\theta)$$

Where θ is in *rad*. Of course ITD is zero when the sound comes from ahead ($\theta = 0$) and it is maximum when it comes from one side ($\theta = \pm\pi$). For a common size of the head ITD is of the order of $10^{-4}s$. This value gives to our brain an important information about the sound source location on the horizontal plane ($\varphi = 0$, *azimuth*). However ITD does not provide the same amount of information on all frequencies. This is because for high frequencies the sound could reach the two ears with a totally different phase and it becomes impossible to define the ITD between two waves that are not in phase. It is commonly used the value of *1.5kHz* as the switching point between a useful and a no more useful information. This value corresponds to the point where the half wavelength become equal (or shorter) than the head size. In addition to these considerations, reverberation can also highly influence ITD: an ambient with a high reverberation decreases the possibility to extract ITD cues, since the ear is exposed to a huge number of reflections coming all at the same time. This is one of the main reasons for which we decided to do not insert reverberation in

2 The head is approximated as a sphere, without losing important information. This assumption is also valid for the rest of this work, for more details see [44].

our experiments. For more information about these aspects, Minaar *et al.* talk about the role of ITD in binaural synthesis [43].

The head also works a shadow to the sound, lowering the *Sound Pressure Level (SPL)* that reaches the farthest ear. The difference in SPL is unnoticeable for low frequencies but it is significant on high frequencies and it is called *ILD* (*Interaural Level Difference*). This is one of the main reasons for which high frequencies play a key roles in the determination of sound source position. In fact, at the contrary of ITD that works better on low frequencies, ILD provides a complementary information on high frequencies and its effect is highly more noticeable than ITD. The cooperation of the two effects is known as *Duplex Theory of Localization*. They work pretty well in the localization on the horizontal plane, but things become more difficult when we should try to estimate the *elevation* of a sound. There is a so-called cone of confusion at about ($\theta = \pm 90$, $\varphi = \pm 30$) where ITDs and ILDs are really close and so it is difficult to distinguish between these positions. In order to correctly identify the elevation of a spatial sound, we need to consider other components.

The role of the external ear

It is not only the head that greatly influences the perception of spatial sounds but it is also the external ear and it has a significant effect on the perception of the elevation. On a first approximation the ear canal behaves like a one-dimensional resonator while the role of the pinna is more complicated and it is still unclear [3]. The theoretical work that provides the basis for our experiments is a possible solution that links the conformation of the pinna to the perception of 3D sounds and tries to answer to some open questions. What it is clear is that the pinna works as a resonator for some frequencies and it causes interferences between the sounds that directly reach the ear canal and the sounds that reflect on its geometric surface. All these effects are directional/distance-dependent so in order to study their role we need to analyze them in function of the sound origin position and this is how the work developed at our Department moved on, see [57]. The key assumption is that at low frequencies the sound signal reflected by the pinna still arrives in phase since the wave length is significantly larger then the distance covered, while on high frequencies the sound signal goes back out of phase and it causes destructive interference. The greatest interference causes the so called *pinna notch* although it is still not clear when it happens. Our studies and our experiments suggest that this happens when the difference is of a full-wavelength [57] even if other previous works suggest that this happens at half-wavelength [50]. More consideration on these aspect will be presented on 2.3.5.

The role of torso and shoulders

Torso and shoulders also influence the sounds that reach our eardrums. As it could be easily imagined, the role of these two elements could be really hard to be considered since the geometry of the torso is quite complicated. However, as we approximated the head with a sphere, it is common to approximate the torso with an ellipsoidal element placed under the head. This system constitutes what is sometimes called the *snowman model*. Some consideration can be highlighted in such a model. First of all, sounds coming from a source on the horizontal plane ($\varphi = 0$) do not vary so much adding the torso model, in particular this happens if their distance from the ear is larger then the head radius. On the other hand the sound delay varies considerably if the sound source is right above or under the subject. The role of the torso and shoulders is not as determinant as the role of the pinna, however it is still important since, because of the size of the torso, it influences the sound at lower frequencies where the response of the pinna is essentially flat.

Distance cues and dynamic cues

There are other factors that contribute to the spatial perception of the sound. For example the estimation of the distance of a sound (that is the most difficult task about spatial hearing) exploits different cues such as the intensity of the sound (even if it is often masked by reverberation) and the nature of the sound (its familiarity with our memories). However there are other aspects that influence the perception of sound distances such as the atmospheric conditions and the curvature of the the sounds waves when the sound source is really close to the ear (at the distance of *1 meter* or less in the so-called *near field*).

One should also consider the possibility of a sound to change its position in the space both because the sound is moving or because the head is moving. These are two different aspects but they both provide a dynamic feedback to the user that greatly improve the localization. It is in fact a common practice to move the head in order to try to better understand the spatial location of the sound.

Front back confusion

Lastly, there is at least one another aspect to be noted about human spatial hearing. It is often noticeable a sort of *front/back confusion* in the sound source localization: sounds that come from a front position are sometimes identify as sounds that come from a rear position. This phenomena (that has been confirmed in our experiments) is quite difficult to be explained. Someone argue that this is an evolutionary aspect: when we are not sure about the real sound position, we place it on a rear position where there is a greater possibility to have an unnoticed and not seen danger situation. Dynamic cues greatly reduce the number of front back confusions [49].

2.3.2 3D sound rendering approaches

Headhones or Speakers?

There are essentially two ways to simulate 3D sounds. The first one uses more than one loudspeaker placed in the ambient, while the second one uses headphones. Both have some advantages and disadvantages. Of course it is quite trivial to generate optimal spatial sounds using a large number of loudspeaker placed all around the listener, however this is a really expensive and unpractical solution. It is widely diffuse with a smaller number of loudspeakers (usually 6 or 8) in the home theater solutions. Some works use instead 2 loudspeaker in a simple stereo configuration and they create the impression of the sound moving from one speaker to the other one, crossfading the signals. However the sound is bounded in the interval from one speaker to the other one. The advantages of these solutions are that the loudspeakers have a flat frequency response and they do not require the listener to wear anything. The disadvantages are that the sounds from each speaker are received always by both ear or it is required a cross-talk cancellation. Also, if there are more then one listener on a quite large ambient, they will perceive the sound coming from relative different positions and the cross-talk cancellation will no longer be possible.

On the other hand headphones systems does not suffer from cross-talking and allow the presence of more listener in the same ambient. The problems with headphones regard the non-flat frequency response (they need to be compensated) and the uncomfortable sensation given by the physical headphones themselves. The binaural headphones synthesis is of particular interest because headphones are cheap and they can reach optimal results. It is a great technology for distributing spatialized music, that could be experienced with minimalistics setups [19]. In this work we focus on the headphones approach.

2.3.3 Head-Related Transfer Functions

HRIR and HRTF definitions

In 2.3.1 we presented the main cues that influence the perception of spatial sounds. Since their effects are linear they can be summed and described by one unique function. In order to introduce this function we consider, for each ear separately, the response of the eardrum to an impulse and we call this response *Hear-Related Impulse Response* (*HRIR*). Its version in the frequency domain (that is its Fourier Transform) is the well-known *Head-Related Transfer Function* (*HRTF*). Experimentally, we can obtain an HRIR be placing a microphone inside the ear canal of a listener and recording the response to an impulse sent from a loudspeaker placed at a particular position at a distance larger than 1 meter (*far field*), than we can perform a FFT on the signal within an appropriate window size to obtain the HRTF. As previously stated, the hearing cues proposed vary hugely with the sound source position so it is ideally required to capture a HRIR/HRTF for each possible sound position. Consequently the HRTF, that is obviously a function of the frequency ω, also depends from the position (θ, φ) of the source and it is formally defined as:

$$H^{(i)}(\theta, \varphi, \omega) = \frac{\Phi^{(i)}(\theta, \varphi, \omega)}{\Phi_f(\omega)}$$

Where i could be both r and l and indicates the right (r) or left (l) ear, and where $\Phi^{(i)}$is the Sound Pressure Level at the eardrum while Φ_f is the Sound Pressure Level placed at the center of the head in a *free field* condition (no listener). Additional information about HRTF can be found on [16], [64]. An individual HRIR/HRTF allows to create a spatial render of a sound coming from the source where the function has been recorded simply convolving or multiplying it with the sound signal after adding the correct ITD. Note that the generated sound will create a correct and perfect spatial sound only for listener who performed the recording of the HRTF (*individual* HRTF).

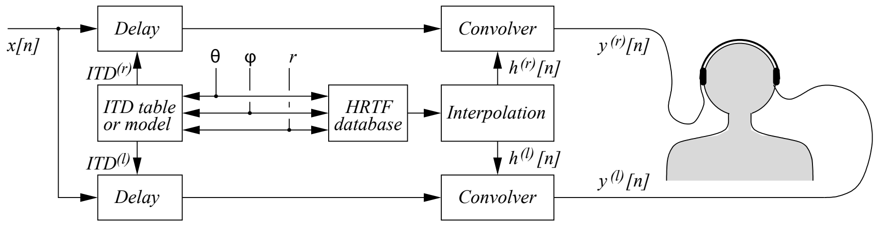


Figure 3: Audio 3D synthesis chain. This system uses binaural rendering with headphones to generate spatial sounds. [2].

HRTF estimation

Since the recording of an HRTF can be limited to a fixed amount of discrete positions and since the procedure of placing microphones near the eardrum is quite invasive, researchers try to obtain reliable HRTF for a particular subject in different and more convenient ways. The first possibility is to use an HRTF recorded on a manikin built upon mean anthropometric data. The most famous one is the *KEMAR (Knowles Electronic Manikin for Acoustic Research*) manikin [21]. However, this approach revealed its limits as we will show on this work (Chapter 5) and as it has been shown on previous works. Another idea is to generate a synthetic HRTF from a physical [38] or structural model that simulates the analyzed components: head, pinna, shoulders and torso. The two main procedures used to achieve this results are the *pole-zero models* and the *series expansions*. Models such these can be really simple [59] or quite accurate and capable to reproduce all the notches and peaks of an HRTF on median plane [58]. All these approach can exploit the principal component analysis (PCA) [33] in order to further tuning the HRTF on the specific subject. Another possibility consist in extracting an HRTF for a subject from a database of existing HRTFs, choosing the “best one” according to certain criteria. In this work we follow this approach.

The CIPIC database

There exist free and open databases of HRIRs/HRTFs. One of the most famous is the *CIPIC* database [1] that we also used in this work. CIPIC is a public domain database of HRTFs extracted from 45 subjects (included KEMAR with small and large pinna). Each HRTF has been recorded at 25 different azimuth positions and 50 different elevations for a total of 1250 positions at approximately 5 degrees of angular increment. The database also contains anthropometric measurement for each subject, pinna images and ITD. Hereinafter we will refer to the subject of the CIPIC database with ID 21 as KEMAR.

2.3.4 Binaural synthesis based on HRTF

Rendering workflow

When the appropriate HRTF (meaning all the discrete recorded HRTFs positions) has been chosen it is quite simple to generate the spatial sound. Figure shows 3 the process. The initial anechoic signal $x[n]$ is delayed by the appropriate amount of time, as indicated on the recorded ITD table (we recall that the CIPIC database provides this information). The HRTF of the required position (θ, φ) is then searched on the database. If the position is not available, then the HRTF is interpolated from the near positions, otherwise the available value is used. The signal is then convolved or multiplied (time or frequency domain) and finally the resulting signal $y[n]$ is sent to the ear. The operation clearly uses different HRTFs for the two ear. In this process eventually *Binaural Room Impulse Response*, *head tracking* and *reverberation* can be considered [5].

HRTF interpolation

We talked about the necessity of interpolate the HRTFs when the required position(θ, φ) is not available. This is because the measurements of HRTF can be made only on a finite set of locations. This is also required to avoid artifacts [2] [19] when the sound changes its position dynamically: since the HRTFs are measured at noticeable intervals (5 degrees in CIPIC) the interpolation is a required step. There are different interpolation techniques. The easiest one is to perform a bilinear interpolation based on the nearest available HRTFs weighted giving larger weight to the closest available HRTFs:

$$\hat{h}[n] = a_1 h_1[n] + a_2 h_2[n] + a_3 h_3[n] + a_4 h_4[n]$$

where $a_i \in [0, 1]\,\forall i$ and $a_i > a_j\,\forall i, j : dist(\hat{h}, h_i) < dist(\hat{h}, h_j)$ However this method has been proved to give biased ITD that still produces artifacts. Another approach uses a virtual loudspeaker environment. One of the most effective method uses instead the decomposition of the HRTF into a minimum phase and a all pass component [30].

2.3.5 Pinna Reflection Patterns and Head-Related Transfer Function Features

Preliminary operations

After the preliminary considerations showed in this chapter, we can now introduce the main results that preceded this work. These results have been achieved by Spagnol, Geronazzo, and Avanzini [57] and are summarized here as an essential reference.

In order to further study the known connections between the pinna reflection/resonances and the elevation cues in sound localization, they removed from the HRTFs of 20 subjects extracted from the CIPIC database, all but the acoustic information of the pinna. In order to discard head, torso and shoulders contribution from the response they mapped each HRTF in the time domain and applied to the resulting HRIR a Hann window of 1 ms since the pinna response lasts from 0,1 to 0,3 ms in the median plane. In this way there is no remaining torso and shoulders contribution. The head contribution to diffraction on the median plane is not relevant so no further modification was made. Going back to the frequency domain gave them the *Pinna Related Transfer Function* (*PRTF*).

Then they analyzed reflections and resonances as two separated phenomena splitting the PRTF into a "resonant" and a "reflective" component exploiting a separation algorithm [22]. They found that, varying the elevation, different CIPIC subjects have similar resonant components trends while there are no common trends on spectral notches, caused be reflections. This gives the idea that for the elevation detection on the frontal region, spectral notches play a key role [45], and so they further analyzed the reflective components.

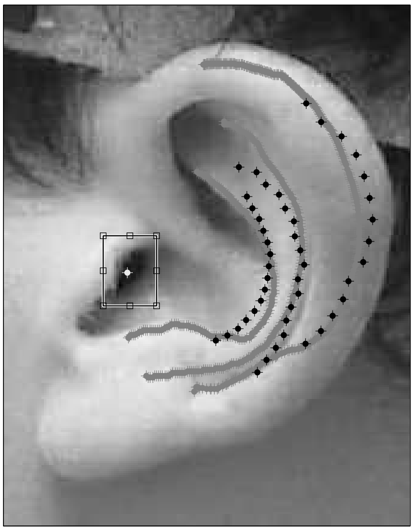


Figure 4: Traced pinna of CIPIC subject 134. Black points represent the extracted notches frequencies, while the lines represents the manually traced contours.

PRTF reflection analysis

To do this, they analyzed the reflective components of the PRTF at 17 elevations [41], starting from +45 degrees to −45° at step of 5.625 degrees. They found that for a given elevation the majority of the 20 subjects exhibits 3 notches. Two subjects lacked of one track of notches. Then they tried to transform the central frequencies of these notches to position on the pinna. To obtain this result they consider the sound with a ray-tracing model instead of a wave-like phenomena. This is possible because the wavelength of the sound is small compared to the dimension of the pinna in the analyzed frequency band so the ray-tracing model is a crude but valid approximation. In this approximation, given an elevation φ, they calculated the temporal delay $t(\varphi)$ between the arrival at the ear canal of the direct sound and the sound reflected by the pinna:

$$d(\varphi) = \frac{c \cdot t(\varphi)}{2}$$

Where c is the speed of the sound and the factor 2 is given by the necessity for the sound to cover 2 times the distance between the pinna reflection area and the ear canal. Substituting the time with the frequency e.g. $f(\varphi) = \frac{1}{t(\varphi)}$, there should be positive destructive interference (a notch) at half-wavelength difference between the two signals, that means:

$$f(\varphi) = \frac{2n+1}{2t(\varphi)} = \frac{c(2n+1)}{4d(\varphi)} \quad n = 0, 1, 2, ...$$

And there should be negative destructive interference at a wavelength difference between the two signals:

$$f(\varphi) = \frac{n+1}{t(\varphi)} = \frac{c(n+1)}{2d(\varphi)} \quad n = 0, 1, 2, ...$$

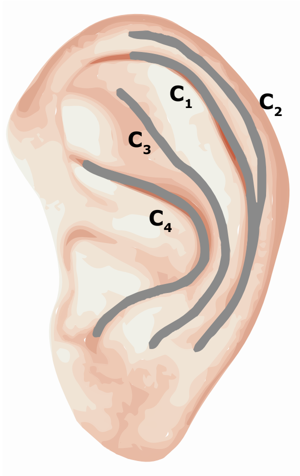


Figure 5: Pinna contours C2/C1 (helixes), C4 (antihelix) and C3 (concha outer borrder).

Starting from the available $f(\varphi)$ they calculated the related distances $d(\varphi)$ with $n = 0$, then for each elevation value, and for each of the 3 notches, they marked 17 points on the pinna image[3], as shown of figure 4 for negative destructive interference. They used the ear canal position (or better a rectangular area as hereinafter explained) as the starting point in the measurement of the distances. They calculated the global mismatch values between the marked points and the manually traced contours as the point-to-point sum of the minimum euclidean distances. In doing this, they chose as ear canal position, the value inside the marked rectangle that minimized the global mismatch. With this mismatch computation process they found that the *negative* destructive model is the one that better connect the position of the notches to the traced contours. So they defined a central frequency f_0 regarded the *negative* destructive interference as: *Results*

$$f_0(\varphi) = \frac{c}{2d(\varphi)}$$

They also gave sindication on the tracing procedure about the three manually traced contours. These contours are shown on figure 5, in particular:

1. C2/C1: helix inner wall and helix border;
2. C4: antihelix and concha inner wall, following the jutting light surface just behind the concha outer border up to the shaded area below the antitragus;
3. C3: concha outer border.

These considerations highlight a connection between the three listed contours of the pinna and the elevation notches. It is consequently possible, given a subject who is not in the CIPIC database with his

3 They obtained the pinna images of the 20 CIPIC subjects, thanks to Professor Ralph V. Algazi.

pinna image, to try to choose the HRTF from the CIPIC database that has the minimum mismatch between the frequencies extracted from the three contours traced on the subject's pinna and the f_0 frequencies of the HRTF. More formally they define the mismatch as:

$$m(T_i, C_j) = \frac{1}{n(T_i)} \sum_{\varphi} \frac{|f_0^i(\varphi) - F_0^i(\varphi)|}{f_0^i(\varphi)} \quad i = 1, 2, 3.$$

where T_i is a track of notches generated from f_0 on a subject in the CIPIC database, C_j is a manually traced contour on the subject who is not in the CIPIC database, $n(T_i)$ is the number of available notches for the *i-th* tracks/contours ($i = 1, 2, 3$).

The psychoacoustic experiment presented in this work compare the results performed in spatial sound localization with a generic HRTF (KEMAR), and two HRTF chosen from the CIPIC database weighting the mismatches of the 3 tracks/contours with different weights and choosing the two smallest mismatches as explained in 3.2.

2.4 HAPTIC BACKGROUND: THE TAMO DEVICE

Introduction to TAMO

Binaural spatial audio is one of the two proposed elements for multi-modal environment in this thesis; the other one is the haptic modality. The haptic device has been chosen to be the *TActile MOuse (TAMO)* device [15] that provides a minimum tactile stimulus. This device, that is still a prototype was developed by the Istituto Italiano di Tecnologia (IIT) at Genua as an aiding device for blind people in the exploration and navigation in virtual environments.

Exploration systems for blind people

The problem of providing a valid aid to blind people in the spatial orientation is a crucial topic. Sighted people use the vision as their predominant sense and all our societies are developed to convey the vision as the central sense. Tasks that look completely easy and natural for sighted people, could become very difficult for a blind subjects, because of their design. The more diffuse device that helps blind people to move and explore the space is the white cane. This element is very simple and lightweight, however it has some limitations: firstly the user needs to effectively be in the space that he or she wants to explore. Secondly it provides only local information about the position and just partial global information, exploiting hearing information in addition to the haptic stimuli. Different alternative solutions have been proposed, however they are often invasive (big and heavy) or they disturb other senses such as the auditory system [20] [13]. *Tactile-vision substitution systems* (TVSS) were the first to be developed and probably remain the most effective since they are the less invasive. One of the most famous haptic device is the Phantom[4], although it is quite expensive, while the TAMO provides a possible future low-budget solution for visually impaired users.

Existing experiments with TAMO

The *Tactile Mouse-shaped* (TAMO) device is part of the DIGEYE®System

4 Phantom's website: http://www.geomagic.com

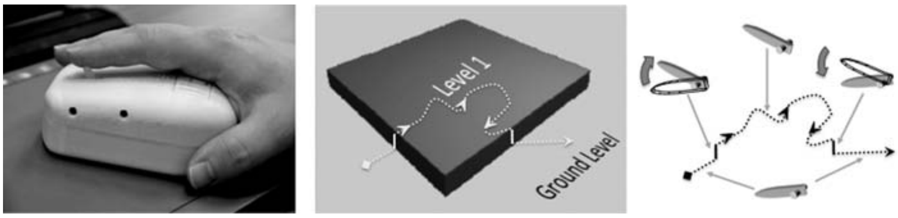


Figure 6: The TAMO device (on the left), a simple virtual map (center), and the movements of the TAMO's lever according to the movements on the map (right) [14]

and it is able to display the height information H at any absolute position (X,Y) of the tablet where it moves [12]. The information about the elevation is provided by a stalk/lever, placed above the mouse-shaped device, in the correspondence of the common position of the mouse-wheel. The lever is progressively moved by a stepper-motor and therefore it provides an haptic feedback to a finger placed above it, as shown on figure 6. A single movement in elevation can be considered a *taxel*: it gives information by one single tactile and it is equivalent to the pixel in the vision[55]. The just noticeable difference in elevation for the TAMO is $JND \in [3.5, 4.3]$ degrees [10]. `DIGEYE` provides a complete test environment for haptic experiments. It was used to preliminary test the TAMO with encouraging results: users were able to correctly identify simple geometric objects and a pilot experiment was also positively concluded on map navigation: a subject was asked to freely explore the virtual map of a room and then to navigate through, avoiding obstacle. The subject succeeded in his task [15]. However, it has been proved that is difficult to reconstruct unknown shapes and to distinguish between them e.g. it is very difficult to discern between a cylinder and a parallelepiped with triangular base, using TAMO.

In this work we would provide preliminary experiments in order to understand the role of audio feedback as an aid to an haptic system based on TAMO, for map recognition and orientation. Instead of using `DIGEYE` we developed a completely new experimental interface based on Matlab[5].

For more technical and implementation-wise information about the TAMO, see 4.2.2.

2.5 PREVIOUS WORKS

Psychoacoust previous works

The last decade registered an increase in the number of psychoacoustic experiment related to HRTFs and binaural audio.

In 2001 Begault et al. [5] performed individual HRTFs extraction, and they tested these functions with head tracking, reverberation simulation (early reflection, full auralization, anechoic) and headphones compensation. They used an input GUI interface quite similar to the one developed by us but their interface looks ambiguous for the choice

5 Matlab® home page. http://www.mathworks.com/products/matlab/, Available: October 2013.

of elevation, a problem that we tried to fix in this work. The achieved results are comparable with ours as shown on chapter 5.

In 2003 Seeber et al. [52] introduced an experiment in order to evaluate the performances of HRTFs extracted from a non-individual group. The aim was to select an HRTF in a two steps process according to multiple criteria such as externalization, mean error and variance. The first step selects the best 5 (on 12) HRTF that achieve greatest spatial perception in the frontal area. The second step does a fine-tune based on 6 direction criteria.

The next year Zotkin et al. [64] performed a complete HRTF selection experiment. However, in a different way from us, they selected the HRTFs by anthropometric data (27 measurements per subject – 17 for the head and the torso and 10 for the pinna). They found an increased performance on 5 subjects out of 6 using selected HRTFs, but they tested only the elevation range from -45 degrees to +45 degrees and bounded the user's selection to the front hemisphere.

In 2008 Shin [53] proposed a procedure for the customization of HRTF based on extraction of median HRIRs, isolation of pinna responses, PCA personal tuning, extraction of pinna responses, head shadow, structural decomposition on median plane to extract shoulder/torso response and individualized ITD. This work achieves really remarkable results in the median plane.

The same year Hwang et al. [28] performed an experiment on modeling HRIR on median plane and comparing them with measured HRIR. The results show that HRIR can be modeled with a reasonable accuracy. This experiment implement an user's input slider interface that is very similar to the one used in our experiments.

In 2009 Douglas et al. [18] presented a method to enhance localization abilities by increasing the spectral contrast of the acoustic features. The final optimized HRTF scores performances that reaches also an increment of 33% on vertical polar localization, compared with the *Kemar*.

In 2012 Katz et al. [32] proposed a method to reduce the size of a 46 subjects database exploiting the quality of binaural synthesis. They found a curious lack of reciprocity in HRTF preferred by the subjects (given 3 subjects *a*, *b* and *c*, if the best HRTF for *a* is the one of subject *c*, it does not imply that this is also the best for *b*). They used LIMSI: a spatialization engine on Max/MSP based on full-phase HRIR convolution. The experiment used an hand user input exploiting a position tracked ball.

Multimodal previous works

Also multimodal systems saw an explosion in the number of proposed works. We try to focus on experiment similar to our and to propose an overview of the existing works.

One of the most famous multimodal virtual environment system for map exploration is *HOMERE* [36] that simulates a white cane: it is used by the subject during the exploration process. However this system does not work with headphones, it uses instead four speakers in a surround system and the subject must explore a predefined path. The system is also quite large and expensive.

Another existing system is *BlindAid* [51] based on the already citied Phantom. This system is designed to allow blind people to explore unknown environment through the interaction with a virtual environment in such a way that is not possible in real worlds. The system is specifically designed to accelerate the user's exploration and learning. This system is able to provide a visual display and record the user's actions in the virtual environment for later review by researchers and O&M instructors. An interesting aspect is that the haptic device Phantom has the ability to simulate textures. However the system does not use HRTF selection but uses a generic HRTF (KEMAR) in order to generate spatial sounds. The authors of BlindAid affirm that a future development will consist in the possibility to render accurate sounds in elevation and consequently moving the spatial sound generation from 2D to 3D. The complete haptic system looks quite complex and it may require a long training process.

Lahav et al. [35] studied the development of a haptic-based multisensory virtual environment that enables the exploration of an unknown space . They also studied the cognitive mapping process of the space by people who are blind working with the multi-sensory virtual environment.

The same authors also performed another similar and preliminary study on map exploration by blind subjects [34]. The main goals of this study were the development of tan haptic virtual environment. The results of the study showed that the participants mastered the navigation in the unknown virtual space with a short time compared to the ones in the control group who had to explore a real space. Significant differences were found concerning the use of exploration strategies, methods, and processes by participants working with the multi-sensory virtual environment, in comparison with those working in the real space. The results were that multi sensory virtual environments are a robust system for the exploration of unknown spaces and for the creation of cognitive maps.

Lederman and Klatzky [37] performed two experiments trying to understand the link between the hand movements and the exploration of objects with haptic feedback and blindfolded subjects. The results establish that in free exploration, the procedure used to explore the object is not just merely sufficient but it is also optimal or even necessary. They also showed that haptic exploration is really efficient in order to explore 3-D objects while it needs to be further developed in order to perceiving spatial layout and structure, presented in the form of raised two-dimensional environments [39].

Walkers et al. [61] analyzed the effects of a nonspeech sound beacon in navigation concluding that the learning of the aiding system improved both speed and accuracy in the navigation.

Since these works obtain different results and since the proposed selection techniques are often long and complex, we decided to perform our experiments in order to test the *pinna reflection model* that consists of a fast and low-cost selection technique.

3 EXPERIMENTS DESCRIPTION

Chapter overview

In this chapter we present an overview of our experiments explaining our choices and motivations. Section 3.1 presents the taxonomy of our experiment and it focuses on the database system that we developed in order to mange subjects. It also presents preliminary auditory tests performed on almost all the subjects. Section 3.2 introduces and describes psychoacoustic experiments while section 3.3 presents multimodal experiments. All the details about the implementation and utilization of the elements described in this chapter, will be presented on chapter 4.

3.1 EXPERIMENTS OVERVIEW

The experiments designed in order to evaluate the effectiveness of binaural rendering and multimodal environments have been split into two main groups:

- Psychoacoustic experiments;
- Multimodal experiments.

The first group contains the experiment created in order to evaluate and to understand the potentials of 3D binaural audio rendered for headphones using a subset of HRTFs extracted from a database. In the second groups there are three experiments evaluate the role of spatial sound in the exploration of virtual multimodal maps.

Subjects Database

Both groups of experiments share a common subjects database created exploiting the *Matlab* `.mat` standard. This approach allows us to create a logical continuity for a subject from the first psychoacoustic experiment to the second group of experiments. The database, that is easily accessible both through a Matlab GUI (see figure 7) and through the Matlab IDE, has been also used in other experiments developed at our Department. Even if it actually contains a small amount of subject, it is designed to be easily scalable in order to contain new data for existing subjects. Now it can manages basic information such as age and gender and it can also manage an image associated to the subject that is his or her pinna image.

3.1.1 Environments

The environments used on psychoacoustic and multimodal experiments are quite similar between them and are based on some shared elements such as (but not only) the common subjects database, implemented in Matlab. Figure 8 shows the psychoacoustic experiment environment while figure 9 shows the multimodal's one. The only differences are the portions related with the TAMO device and the

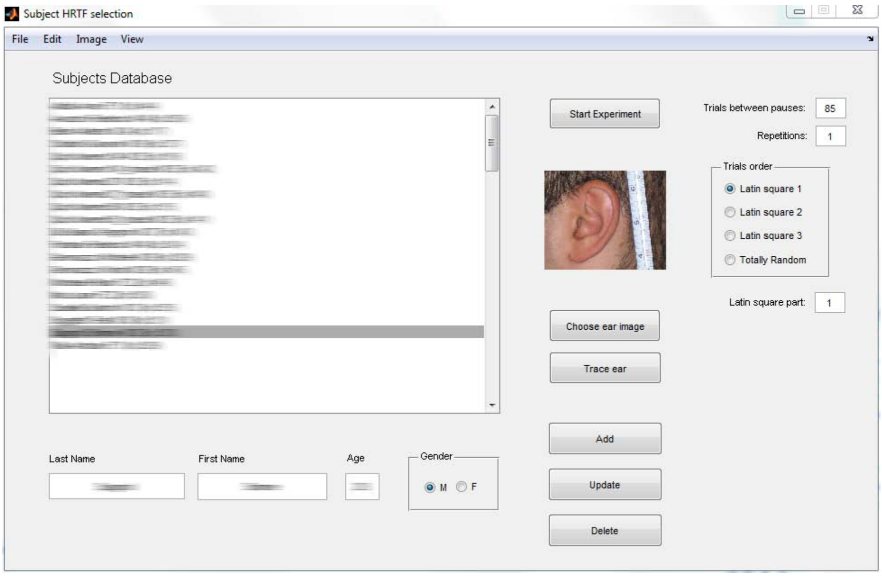


Figure 7: The database GUI containing all subjects. This GUI also manages psychoacoustic experiments and it is really similar to the one that manages multimodal experiments.

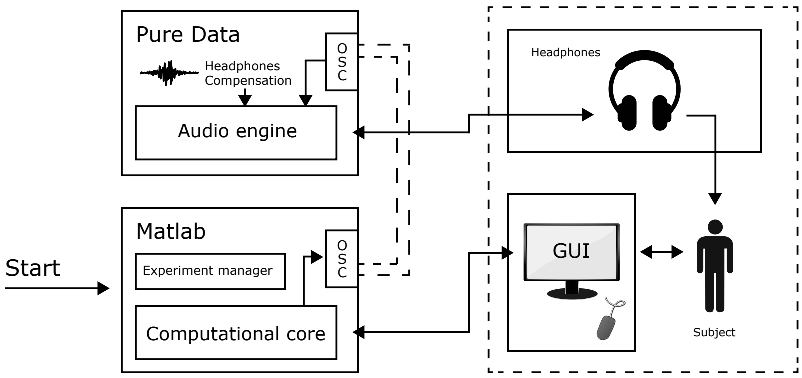


Figure 8: Psychoacoustic experiment chart setup

way of how the subject interacts with the system. On the psychoacoustic environment he or she interacts in a quite intuitive way: giving feedbacks/answers on a graphic interface on the screen and using a common mouse, while on the multimodal environment he or she must actively interact using only the TAMO device. On both experiments audio signals are processed by Pure Data[1] using the *Cw binaural* audio engine and sent to the subject through headphones. Sound to be processed are sent to Pure Data by Matlab using the *OSC*[2] protocol. The experimenter manages and coordinates the whole systems from Matlab. This section describes these software and mechanisms.

1 Pure Data website: http://www.puredata.info
2 OSC webiste: http://opensoundcontrol.org

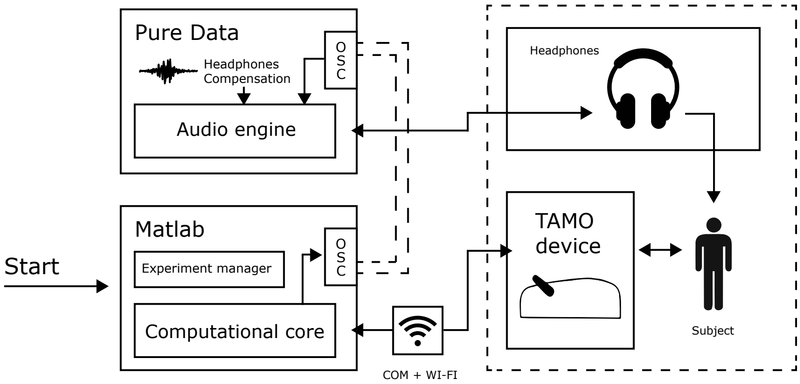


Figure 9: Multimodal experiment chart setup

3.1.2 Auditory test

Preliminary auditory test approach

In order to perform our multimodal and psychoacoustic experiments, we need to certified that all our subjects are able to properly hear sounds at all frequencies until 8 kHz (standard audiometer range) according to a reference dB level. To achieve this result, Sandro Scaiella[3] developed a preliminary test based on "*MLP: A MATLAB toolbox for rapid and reliable auditory threshold estimation*" [24]. This test has been proved to reach a good degree of accuracy and in a short time. It allows auditory threshold estimation via the *adaptive maximum likelihood* (ML) procedure proposed by David Green [25]. The ML procedure is largely known, used, and appreciated by the auditory community because it can exploit yes-no questions instead of forced-choice procedures. The main advantage of yes-no questions is that they are more intuitive and the subject does not need to know all the possibilities before giving an answer. This is really effective in our context where we already have a large amount of tests to perform and we do not want to tire the subjects. As reported by Green, the standard deviation obtained in these tests using 12 trials[4] (we used 20) is 2-3dB, that is enough to identify any hearing problems. Subjects with these sort of problems have been rejected. It could happen to find subjects who do not know that they a problem hearing sounds in a particular range of frequencies. As an example and reference we reported on figure 10 the results of an auditory test. Please note that there is a deterioration of performances at around *2000* and *3000 Hz* due to headphones non linearity above and at these frequencies (see 4.2.1 for more details).

3.2 PSYCHOACOUSTIC EXPERIMENTS

This group of experiments could be consider as just one large experiment that evaluates 3D binaural audio rendered using an HRTF database.

3 MS Thesis Student, Department of Information Engineering, University of Padua
4 In this context a trial is an audio stimulus at a particular dBA level.

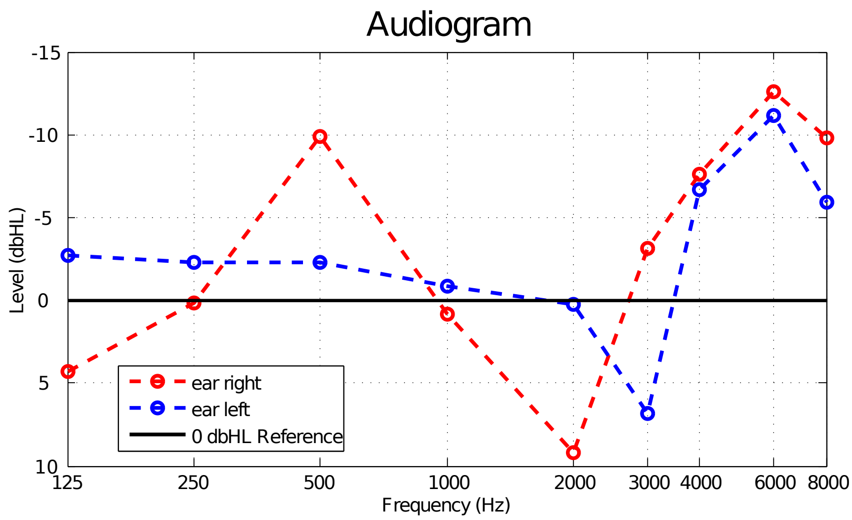


Figure 10: An example of an audiogram output.

Pinna image manipulation

As explained in 3.1 each subject inserted into the database is associated with the image of his or her external left ear. We developed a GUI that allows the experimenter to manipulate this pinna image. On chapter 2 we said that pinna contours greatly influence the way we perceive spatial sounds (in particular the elevation) so it is critical for us to be able to efficiently extract important data from these images. In particular we need to:

- Crop and scale pinna images;
- Rotate pinna images in order to horizontally align the tragus point to the most external point of the subject nose;
- Manually trace the three critical contours using a pen tablet;
- Pick a focus point (ear canal);
- Define a scale factor from a ruler impressed on the image.

In order to achieve these results, we created an effective GUI that allows to draw the three contours using a pen tablet: since tracing good contours require a bit of practice, we tried to minimize the difficulty of this operation providing an intuitive input interface on a really large tablet. Also GUIs were created to graphically straight the image (drawing a reference line between the tragus and a nose point) and to pick up a scale factor just drawing a line one centimeter long above the ruler on the image. The traced contours are then used to calculate scaled distances from the focus point and consequently reflection patterns and frequencies as previously described. These frequencies are then stored in our database.

3.2.1 Pinna images acquisition and trace

Environment setup for pinna acquisition

Pinna image acquisition is the first step performed in order to add a new subject to the database. We created a controlled setup in order to capture pinna images on different days. The setup is shown on figure

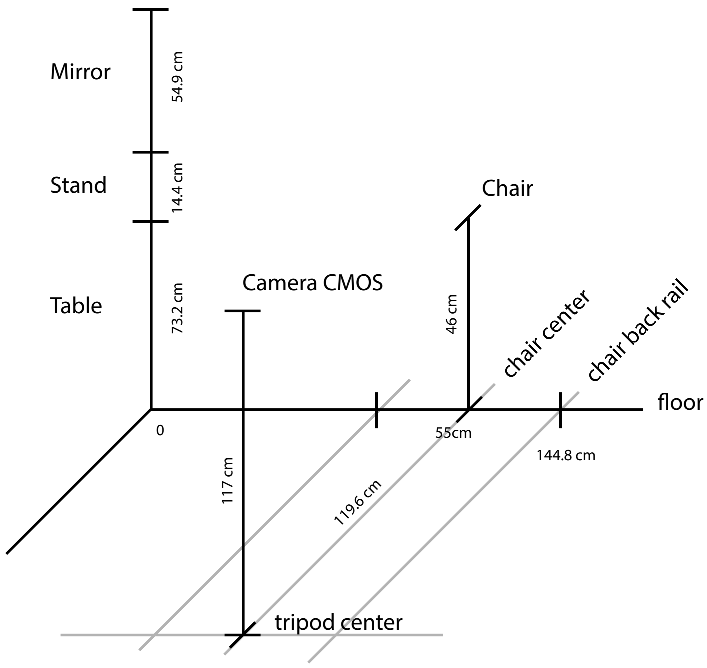


Figure 11: Pinna acquisition setup

11 with the element relative positions involved in the process. We asked the subject to sit down on a chair and to center his or her head on a mirror placed parallel to the subject. In the center of the mirror, we placed a straight vertical line: we ask the subjects to alternatively close one eye and to move their head in order to superimpose the other eye to the vertical line. In this way we avoid unwanted head rotations on one dimension (left and right head pan). The second dimensions (tilt of the head up and down) is fixed on post-production aligning the image according to specified parameters (see 3.2). The remaining dimension is checked using a level. The mirror is placed on a support on a table. The image of the pinna is then captured using a camera placed on a tripod. The camera CMOS sensor is perpendicular to the pinna of the subject. Even if it is not shown on the image, the setup is symmetrical so there is the possibility to capture both pinna images. We ask the subject to move his or her hairs in order to clearly shown the entire ear/pinna. We also place a ruler near the ear of the subject in order to have a fixed pixel to meter conversion factor. The images were captured using a Canon Digital IXUS 85IS at 18.6mm of focal length (35mm equivalent: 105mm). Each image is 10 MegaPixels in format 4:3. The camera at 18.6mm is using its maximum optical zoom factor, in order to reduce as much as possible the lens distortions.

Image processing

The captured image is then processed in Matlab as previously described. All the processing operations can be performed using an ad-hoc created GUI that exploits some of the main feature of the Matlab library from both standard and image processing toolboxes.

Scale

The first operation performed on captured images is scaling. We

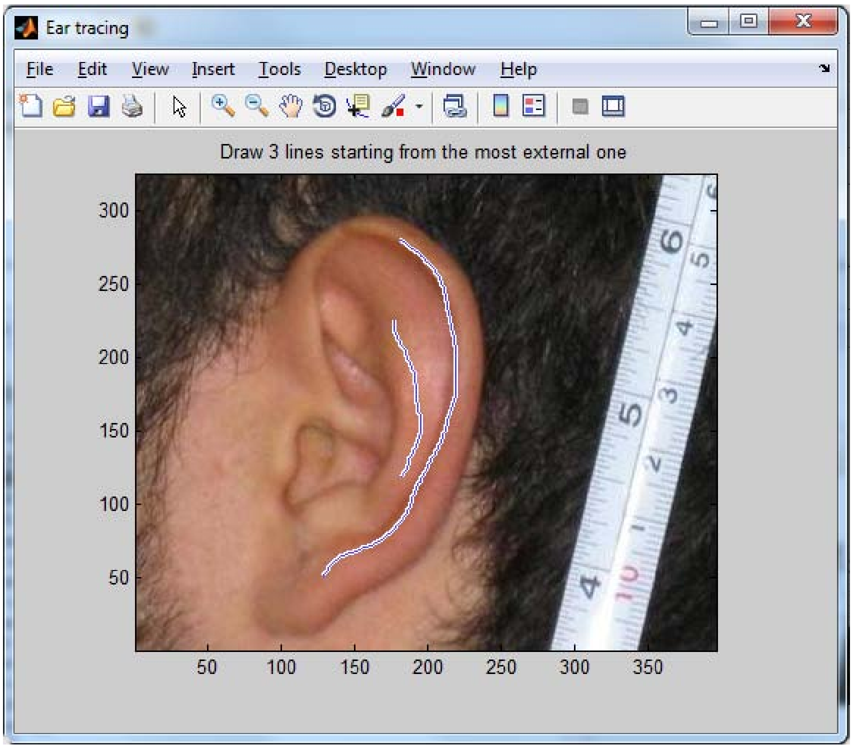


Figure 12: Pinna tracing. The experimenter is tracing the second contour.

actually scale the image at 50% of its original resolution (recalling it is a 4:3 10MP image) using bicubic interpolation: the output pixel value is a weighted average of pixels in the nearest 4-by-4 neighborhood.

Rotate

Then we proceed rotating the image: the experimenter traces a line from the most external point of the subject's nose to the tragus of the pinna image and we extract the rightmost angle between this line and an arbitrary horizontal line. If the angle is larger than 90 degrees, the image is rotated in CCW, while if it is smaller the image is rotated in CW. On both cases the rotation is performed using bilinear interpolation. Finally the experimenter simply crops the image in order to contain just the pinna and a small amount of the ruler.

Pixel to Meter factor

The last step is to acquire the pixel to meter factor. Since we have no way to capture images with a 1:1 reproduction factor lens, placing a ruler on the image is the easiest way to measure real absolute distances. The experimenter must trace a straight line on the image covering the distance of 1 cm on the ruler. The software then computes the euclidean distance between the endpoints of the traced segment, in pixels, and define a pixel to meter factor as `0.01/distance`. This value is automatically stored on the clipboard and it is asked at the end of the contour trace phase.

Image tracing

The experimenter must now trace the three critical contours of the pinna image. This operation requires some practical experience and it is critical in order to correctly match the HRTFs from the CIPIC database. We implemented a GUI showed on figure 12 that allows an easy tracing process. The experimenter starts tracing the most external contour, then he or she continues with the middle one and finally the most internal one. Then the experimenter must place the ear canal position with a single click from the GUI above the image. The last step is to paste the pixel to meter factor previously stored. Contour trace operations use the library function `imfreehand`.

The experimenter, instead of the mouse, traces using a large (more then 50cm) *Wacom* tablet that allows precise movements. The final data are stored in a `.mat` file associated to the subject's image name. It contains: `Cx` and `Cy`, two 3 columns vectors containing the raw traced points of the 3 contours, starting from the most external one; `earCanal` that is the position coordinates (x and y) of the ear canal; `F0`: a 3x17 vector that contains the frequencies extracted from the image as described a pagina 53. Each row corresponds to a contour, again starting from the most external one, while each column corresponds to a angle value from +45° to -45°; `dist`, a vector the same size of `F0` that contains distances instead of frequencies; and `pixelToMeterFactor` that is the conversion factor calculated as described in this paragraph.

Images are always stored as `.jpg` files in the *img* directory, keeping a backup copy of the original unedited image. Contours are stored as `.mat` files in the *img* directory.

3.2.2 HRTF selection

HRTFs selection

The stored frequencies values are then used to select HRTFs for the experiment. Actually we store 17 values per contour for a total of 51 values (there are 3 contours). Each value is associated with the distance between the focus point and a contour for a particular angle (elevation) value, starting from +45° to -45° at step of 5.625°. Then for a particular subject we choose the two HRTFs that minimizes the frequencies gap (*mismatch*) with the subject stored frequencies values. These two HRTFs are chosen from a database weighting with different weights the frequencies values associated to the counters and avoiding duplicates. The database used in our experiment is the *CIPIC* database (see [1]). We also choose a fixed third HRTF that is the one associated with the *KEMAR*[5] test subject (`CIPIC ID 21`). The idea is that the two chosen HRTFs should render better spatial sounds than the KEMAR's one due to the relation between spatial sounds and pinna cues. The mismatch is calculated in the following way: for each CIPIC subject we calculate the differences (element by element) between the 17 frequencies values of the current CIPIC subject and the 17 values of our subject, then we sum them. We repeat the operation for each contour and then we sum the 3 mismatches with different weights: for one criteria we use the weights $\frac{1}{3}$, $\frac{1}{3}$, $\frac{1}{3}$ (*criteria 2*) and for the other one (*criteria 3*) we use the weights 1, 0, 0. The first criteria (*criteria 1*) is bounded to choose the subject from CIPIC with ID 21 (KEMAR), as a reference and comparable subject. The previously listed weights corresponds to the weight of each contour in the sum of the mismatches, starting from the most external contour. For example, calling $mismatch_1$ the mismatch of contours `C1/C2`, $mismatch_2$ the

5 Knowles Electronics Manikin for Acoustic Research, see [1]

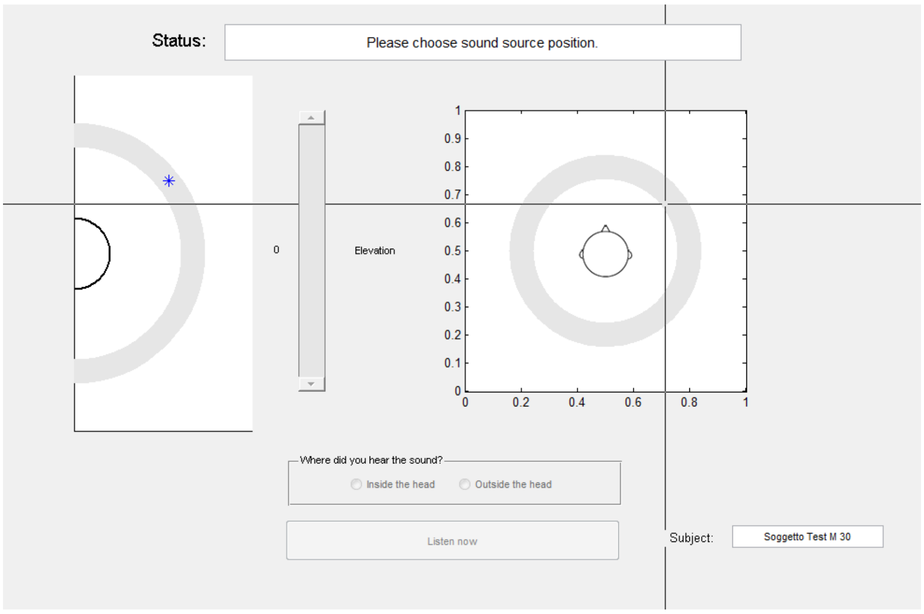


Figure 13: The GUI for the psychoacoustic experiment.

mismatch of contour C4 and $mismatch_3$ the mismatch of contour C3, the total mismatch is computed as:

$$mismatch = weight_1 \cdot mismatch_1 + weight_2 \cdot mismatch_2 + weight_3 \cdot mismatch_3$$

Then the lowest mismatches are selected, avoiding duplicates. In 4.3 we described this algorithm.

Apparatus

The experiment apparatus is described in chapter 4. It should be noted that we decided to provide an experimental setup on average and not-optimal conditions e.g. we do not use personal headphones compensations. This is because we would like to simulate a possible application scenario where it is impossible to measure personal compensations.

Procedure

Running the experiment

After the selection phase of HRTFs, the real experiment can start.

The subject is instructed on how the experiment works, giving him or her the less possible amount of information needed in order to complete the experiment with success. In particular the subject is instructed on how the experiment GUI interface works and a brief tutorial of the experiment is run. This tutorial is run both to allow the subject to become familiar with the interface and to allow the experimenter to control that the subject correctly understood the protocol and that everything is working as expected. During the tutorial some trials (usually from 10 to 15) are presented with a generic HRTF. The experiment metaphor and protocol are explained with the following standard sentences:

Figure 14: The silent booth during the psychoacoustic experiment with a subject.

- "Spatialized sounds will be played on your headphones";
- "You should try to identify the sound source location";
- "Move this slider and place it on the upper half of the bar if you think that the sound comes from above you, otherwise place it on the lower half";
- "Placing the slider in the upper position represent a sound coming from above your head; placing the slider in lower position represent a sound coming from under your body. Note that you can also leave the slider at its original position when you think that the sound comes from the level of your ears. You can also check the current selected position watching the reference image on the left";
- "Then place a point on this circle, in the direction where you think the sound is coming from";
- "Finally specify if you hear the sound coming from inside or outside your head";
- "Press continue and repeat until the experiment completes or a pause pops up".

Experiment GUI

The experiment take place in a silent booth, see figure 14. During the experiment a spatialized sound noise impulse trains are generated and sent to the subject headphones. These sounds are generated convolving the impulses with 3 chosen HRTF, as explained in the next paragraph. The subject must choose on the GUI (see figure 13) what he or she thinks that is the real source of the sound (both in elevation and azimuth). We decided to place the elevation slider before the azimuth graph since for us elevation results are more important than azimuth results. This is because for a subject it is harder to identify the

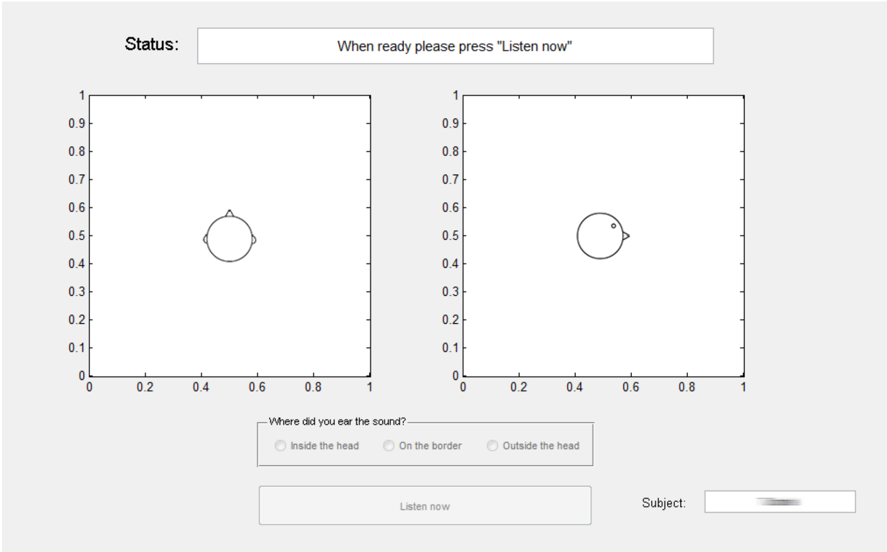


Figure 15: The GUI for the psychoacoustic experiment (old version).

elevation than the azimuth [19] [7], and we also expected that HRTF selection significantly influences the elevation results. The subject must also choose the sound source position with radio buttons: did he or she hear the sound inside or outside his or her head? In this way we also collect data on externalization. We would also like to underline that the current GUI is the result of a continuous series of changes as a consequence of sequential evaluation which goal was to iterate towards a better user interface for a particular application [9]. At the beginning there were two graph in which the subject could choose the azimuth (in the first one) and the elevation (in the second one). This brought to a series of possible incoherent and no intuitive entries for elevation: see figure 15. That interface was really similar to the one used on [5]. We performed some pilot experiments with experts subject in order to identify the best possible input interface. A slider looks like an intuitive way to choose an elevation value. The slider maps all possible elevation values from a minimum of -90° to a maximum of +90° using the `arcsin()` function. As an additional graphic confirmation we added a point that moves on a semi-circle and that corresponds to the current selected elevation. In addition on the first version of the GUI there were the possibility to choose from a radio button a value from 1 to 5 that represented how much the subject was sure about his or her choice. Also the externalization was defined as a value in a scale from 1 to 5. However, the experiment resulted too long and challenging so we decided to gradually simplify the interface. The final interface provides the possibility to freely decide all possible positions on an virtual sphere where the subject is the center of the sphere. This is the main reason for which we decided to do not use a finger point tracker system that does not easily allow to point positions on the back hemisphere.

Stimuli

Each stimulus is a train of 3 sequences of white noise impulses at 60 dbA measured at the pinna level on the headphones. Each single sequence consists of 3 impulses which last 40 ms per impulse. The impulses are separated by a pause of 30 ms. Each sequence is separated from the next one by a pause of 320 ms. This type of stimulus has already been proved to be effective and better than a simple white noise [32]. It is known that "sounds having energy for all of their frequency bands, such as white noise, are easier to localize than sounds having a sparse spectral content (sine tone)" [19]. The order of sound stimuli can be decided before the beginning of the experiment and it can be set to "random" or "latin square" order. See 3.4 for more informations about this aspect. The experimenter can also set the number of repetitions of each stimulus that is the number of time that a particular stimulus is presented at the same position. Since the experiment could be quite long we decided to add pauses at regular intervals between trials.

In our experiments we presented to the subject 85 sound positions per HRTF. Each position is repeated 2 times. Since we use 3 HRTFs there is a total of:

$$85\,positions \cdot 2\,repetitions \cdot 3\,HRTFs = 510\,stimuli$$

The positions are generated using all possible combinations of these values of azimuth: [0, 30, 60, -30, -60] and these values of elevations: [-45, -30, -15, 0, 15, 30, 45, 60, 90, 120, 135, 150, 165, 180, 195, 210, 225] both expressed using vertical polar coordinates. In order to provide some rest to the subject, we added 1 pause of 3 minutes after a single block of 85 positions. We measured the length of an experiment and it is of about 70 minutes, pauses included. The order of the stimuli is widely discussed on 3.4.

A compensation was performed on headphones as described in 4.2.1 so the final stimulus is therefore the convolution of the train of impulses, headphones compensation and selected HRIR.

Subjects

ID	Age	Gender	Lat. Sq.
17	24	M	3
29	29	M	1
30	28	M	2
36	40	M	1
43	26	M	1
44	31	F	2
46	22	M	3
48	22	F	2
50	27	-	-
51	27	-	-

Table 1: Psychoacoustic Subjects. Subjects with ID 50 and 51 are fictitious subjects created as a reference as explained in 5.1.1. The latin square value was assigned as explain in 3.4.

Subjects overview

Table 1 contains the list of all the subjects who performed our psychoacoustic experiment. There is a total of 8 real subjects, 6 male and 2 female. The age varies from a minimum of 22 years to a maximum of 40 years with a mean value of 27 years. Three of these subjects (ID 29, 30 and 36) had already sustained psychoacoustic experiments, the other was naive subjects. These considerations are important because we found a learning curve in the experiment as explained in chapter 5. We provided the subject with the minimum amount of the required information in order to perform the experiment. After the brief tutorial section, they were left alone in the silent booth in order to allow them to focus on the task, without distraction. The experimenter was able to monitor the experiment from outside the booth, without to be seen. Some subjects preferred to close their eyes during the experiment. Some preferred to turn off the light, some preferred the light turned on.

Results

The results are then stored in the database and they are automatically parsed in order to extract significant data. In particular the performances of the 3 different HRTFs are parsed in different spatial ranges. Results are presented in chapter 5.

3.3 MULTIMODAL EXPERIMENTS

Multimodal protocol introduction

In these experiments we analyzed how 2D and 3D sounds could help the exploration of virtual spatial maps. With 2D audio we consider only variation on azimuth with respect to the position of the subject, while with 3D audio we also consider elevation changes on the virtual map. This group of experiments uses the HRTF that scored the minimum mismatch between real sound positions and chosen positions extracted from the previous experiment. The HRTF is used in order to generated the best possible spatial sound for the particular subject. Note that in order to perform these experiments it is not needed to perform the psychoacoustic experiment as a whole: we just need to acquire the pinna image of the subject and to calculate mismatches with the contours, then minimize them according to a particular criteria as explained on 3.2.2. The psychoacoustic experiments evaluate the quality of this minimization criteria and we reputed it sufficient for these experiments.

This group of experiments is composed by 3 experiments:

- Goal reaching (experiment 0);
- Object recognition (experiment 1);
- Spatial map reconstruction (experiment 2).

Multimodal experiment metaphor

In all of these experiments the subjects are unable to see (congenital or forced using an eyes-mask). They are told to explore a virtual spatial map and to obtain a particular result: reach a position, reconstruct an object or memorize the map. In order to achieve these results, they can explore the map using the TAMO device and they can exploit the presence of a global sound source generator placed on a particular spot. The audio can be both 2D and 3D rendered. The experiments are also repeated without the sound in order to evaluate multimodal and mono-modal contributions. The subjects must think to be the mouse-pointer (TAMO) even if they do not see it. The spatial sound is rendered according to the relative position of this pointer to the position of the sound marker. The TAMO reacts to the presence of an object by moving its stepper (lever): when the subject (the mouse pointer) is above an object on the map, the stepper goes up and vice versa. . When the subject places the devices outside the map, the lever repetitively moves up and out in order to signal an error situation: this simulates a virtual wall. The experiments are explained to the subjects with particular standard sentences that describe the metaphor of the experiment itself. These metaphors are listed below, one for each experiment. Maps are very simple rectangular spaces containing one geometric figure such as a cube (see figure 16). Some maps are surrounded by a virtual wall that corresponds to the higher position of the TAMO lever. This wall works as a reference point for the end of the map in the experiments where there is no sound. In fact in the experiments with audio, the sound stops when the subject goes outside the map.

Apparatus

As for the previous group of experiments, the experiment apparatus is described in chapter 4.

Stimuli

Subjects can receive two stimulus: the haptic one, that is a lever that moves up and down creating a pressure on a subject's finger. The lever

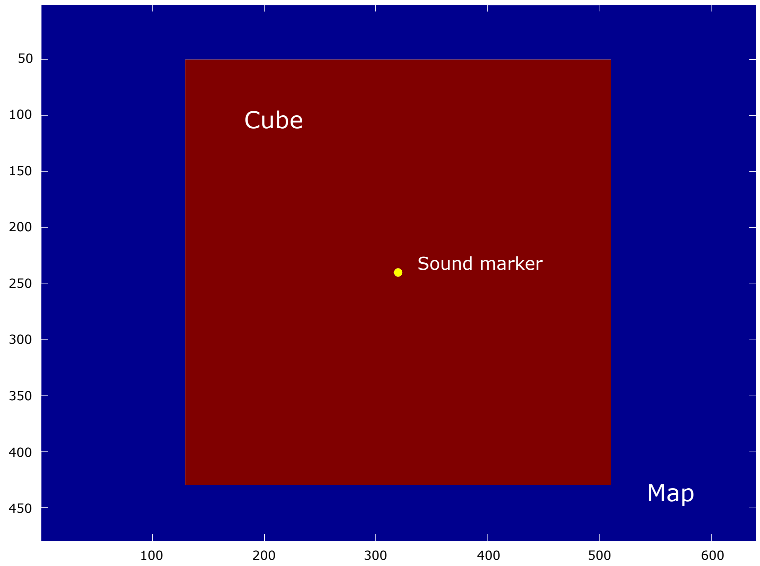


Figure 16: This is a sample map for multimodal experiments. It is a 2D top view of a cube at the center of the rectangular map. It is possible to recognize the sound marker source position at the center of the map.

is placed at the middle of the TAMO device, in the same position of a common mousewheel. The lever moves from a ground position of horizontal 0 degrees (TAMO surface) to the maximum position that is near the vertical position and correspond to a rotation of about 80 degrees.

The second stimulus is the sound reproduced by the headphones that corresponds to the spatial sounds generated by the audio marker. It is a series of continuous white noise impulses that last 40 ms each one at a default level of 60 dbA measured at the pinna level on the headphones. At the beginning of the experiment the subject can decide to increase or to decrease the volume of the sound in order to reach a comfortable level. Noise stimulus has been proved to be the most efficient in these experiments [61]. The sound pressure level is the same for all experiment but it can be further lowered on the last one. The reason for this is to increase the focus ability of the subject on his or her main task. When the TAMO device is moved outside the virtual map, all sounds stop as a warning condition.

Subjects

ID	Age	Gender	Lat. Sq.
29	28	M	2
30	28	M	2
36	40	M	3
43	37	M	1 (pilot)
44	26	M	1
50	30	F	1
51	23	F	2
52	24	M	3
53	27	M	3
54	26	M	3
55	21	F	2
56	28	F	1

Table 2: Multimodal Subjects. The latin square value was assigned as explain in 3.4.

Subjects overview

Table 2 contains the list of all the subjects who performed our multimodal experiment. There is a total of 12 subjects (pilot included), 8 male and 4 female. The age varies from a minimum of 21 years to a maximum of 40 years with a mean value of 28 years. Two of these subjects (ID 29 and 36) had already sustained multimodal experiments, the other was naive subjects. These considerations are important because we found a learning curve in the experiment as explained in chapter 5. We provided the subject with the minimum amount of the required information in order to perform the experiment. After a brief tutorial section, the experimenter drove them through the experiment remaining inside the silent booth.

3.3.1 Goal reaching (experiment 0)

Procedure

Experiment description

This is a preliminary experiment created in order to verify the attitude of subjects to reach a specific position: the center of the map. The sound marker is placed at the center of the map and the pointer is place on the border of the map as shown on figure 17. The subject, following the sound generated by the marker, should try to reach the exact position of the marker, placing himself or herself above it and to hold the position until the experiment automatically completes. The subject must remain near the marker for 1.2 seconds. The minimum distance to trigger the timer is 25px (recalling that the map is 640x480 pixels). The experiment is introduced with the following statements:

- "In this experiment you will use headphones that play sounds coming from a fixed source";
- "You will be blindfolded and you should try to explore a map and to reach the sound source as fast as you can";

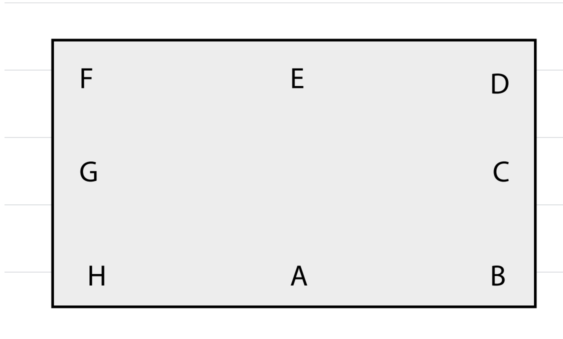


Figure 17: Multimodal Exp 0 starting positions. This figures shows the starting position for the first multimodal experiments.

- "In order to explore the map, you have to move this device, as if it was a mouse";
- "Try to think that you are the mouse: your position on the map is the mouse position";
- "Keep you head ahead";
- "On the top of mouse there is a lever that will move up touching your finger. The more you will go closer to the center, the more the lever will move up";
- "You can think as there was a hill at the center of the map: the lever goes up as you go up on the hill";
- "You will repeat the experiment with both audio and lever, with only the lever and with only audio".

The map is generated with the haptic elevation that grows near the center with an inverse square law as shown on figure 18. The experiment is repeated 8 times (one for each starting position) for each of the 3 audio/haptic conditions (only TAMO, only Audio, TAMO and Audio) in order to compare the different performances and to see if the two modal systems work well together.

3.3.2 Object recognition (experiment 1)

Procedure

Experiment description

The subject must explore the map trying to identify the object created in the virtual map in the less time he or she can. The object is placed in the center of the map and so the sound marker. There is a maximum amount of time of 150 seconds. For each object, the experiment is repeated two times: one with 2D sound and one without the sound.This experiment is presented in the following way to the subject:

- "In this experiment you have to explore a virtual map";
- "In order to explore the map, you have to move this device, as if it was a mouse";

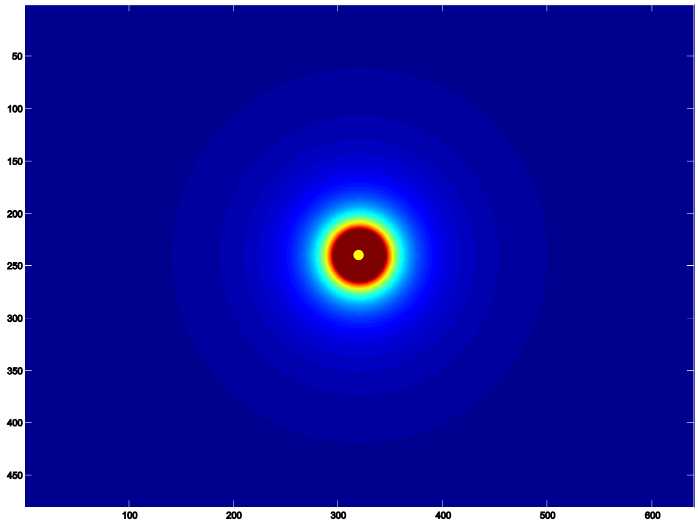


Figure 18: Multimodal Exp 0 map. Haptic elevation is shown in color and it increases near the center.

- "Try to think that you are the mouse: your position on the map is the mouse position";
- "There is a geometric object placed in the map.You must try to identify this object as fast as you can";
- "When you will place yourself (the mouse) above the object, the mouse lever will move up, otherwise it will go down";
- "You will repeat this experiment both with and without audio and with different geometric objects";
- "When you will hear the audio, it is coming from a sound source placed at the center of the object";
- "It is important that you understand that, at the contrary of the previous experiment, your objective is not to reach the sound source: audio only helps you to understand where you are on the map";
- "As soon as you identify the object, tell me your guess: the experiment will complete when you correctly identify the object or after a maximum fixed amount of time";
- "When you give an answer, you must be quite sure about it".

With this experiment, we tried to understand if the sound effectively helps exploring the map. The objects are:

- Parallelepiped with square base;
- Parallelepiped with triangular base;
- Cylinder.

Experiment outcome The subject can try to guess the shape of the object multiple times and the experimenter does nothing when the answer is incorrect and he or she stops the experiment when the answer is correct. With the last 4 subjects we also noted the times and the wrong answers given by the subjects. We also see that in this type of experiments it is quite common to ask to the subject "Are you sure?" before confirming the correctness of the answer: sometimes the subject continues the exploration a bit more. However we are not able to interact with the subject while he is hearing sounds and since we are interested in performance times, removing the headphones during the experiment or lowering the volume is not an acceptable solution. If the subject is not able to identify the object on a large fixed amount of time, the experiment is concluded with a negative outcome.

Object sizes An important aspect of this experiment is the virtual objects size on the map. Using pilot experiments we found out that the objects should occupy a large amount of the map so they consequently also occupy a large amount of the physical tablet. This was also confirmed by [11]. In order to achieve this result the TAMO device should physically move from about 10cm to 15cm remaining above the virtual object. With this constraints the shapes must be larger then 50% of the tablet e.g. we created the cube a side of 380px (recalling that the map is 640x480 pixels). However these values depend both from the screen resolution (in pixels) of the computer and from the video card of the running machine and from the mouse/device speed. Additionally, the height of these objects is set to 18 degrees of the lever of the TAMO device calculated from the horizontal zero level (TAMO surface).

Appendix A contains all the map used in this experiment.

3.3.3 Spatial map reconstruction (experiment 2)

Procedure

Experiment description The subject must explore the map trying to understand the position and the size of a cube randomly created in the virtual map. The sound source is placed in the center of the map.This experiment is introduced in the following way to the subject:

- "In this experiment you have to explore a virtual map";
- "In order to explore the map, you have to move this device, as if it was a mouse";
- "Try to think that you are the mouse: your position on the map is the mouse position";
- "There is a cube on the map: it could have different sizes";
- "You have to explore the map and search for this cube. Try to memorize its position on the map";
- "You have also to try to guess the size (length) of the cube";
- "When you will place yourself (the mouse) above the cube, the lever of the mouse will move up, otherwise it will go down";

- "You will repeat this experiment both with and without audio and with cubes of different sizes".
- "When you will hear the audio, it represents a sound source placed at the center of the map";
- "It is important that you understand that, at the contrary of the first experiment, your objective is not to reach the sound source: audio only helps you to understand where you are on the map";
- "You have 1 minute to explore the map. When the time ends up, you will have to choose one of these 5 cubes and to place it on a rectangular space as large as the explored map according to the memorized position and size";
- "It is possible that on the entire experiment there are not all the 5 cubes sizes".

For each cube size, the experiment is repeated three times: one with the 3D sound (exploiting the HRTF selection), one with 2D sound and one without the sound. We tried to understand if the sound effectively helps exploring the map. The experiment is also repeated with cubes of 2 different sizes: small (80px) and big (150px). The subject must choose between the following 5 sizes: 45px, 80px, 115px, 150px, 185px. Each cube/audio configuration is repeated two times for a total of:

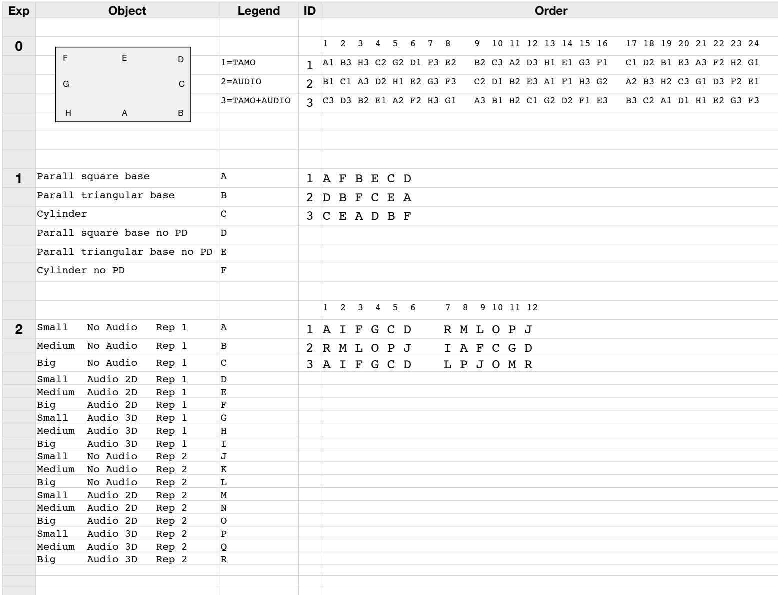
$$2\,sizes \cdot 3\,audio\,conditions \cdot 2\,repetitions = 12\,stimuli.$$

3.4 TRIALS ORDER: LATIN SQUARE AND RANDOM

The order of the trials is really important in these experiments because the learning curve for a subject who has no experience with multimodal application and 3D binaural audio can be dramatically fast-growing at a first stage. In order to reduce this effect, as previously described, a tutorial is performed in order to give the subject a first contact with the interface and the metaphor. However the real experiment introduces a lot of practical experience so we must take into account this aspect. If we want to compare performances in different situations (selected HRTFs against KEMAR, and audio against no audio) we must present these situations both to the same subject and to different subjects with different orders.

Random order

The first and most intuitive way to obtain this result is to present stimulus in a total random order. On multimodal experiments this approach could be quite effective with a large number of subjects that minimizes the probability to present a group of similar stimuli or the same stimulus all together for all the subjects. However these experiments are not planned to be performed on a such large number of subjects so a random order may not be the best choice. On psychoacoustic experiments a total random order would mean to present sounds with different HRTFs shuffled together. This could causes a lot of confusions to the subject affecting the final results.



Exp	Object	Legend	ID	Order
0				1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
	F E D	1=TAMO	1	A1 B3 H3 C2 G2 D1 F3 E2 B2 C3 A2 D3 H1 E1 G3 F1 C1 D2 B1 E3 A3 F2 H2 G1
	G C	2=AUDIO	2	B1 C1 A3 D2 H1 E2 G3 F3 C2 D1 B2 E3 A1 F1 H3 G2 A2 B3 H2 C3 G1 D3 F2 E1
	H A B	3=TAMO+AUDIO	3	C3 D3 B2 E1 A2 F2 H3 G1 A3 B1 H2 C1 G2 D2 F1 E3 B3 C2 A1 D1 H1 E2 G3 F3
1	Parall square base	A	1	A F B E C D
	Parall triangular base	B	2	D B F C E A
	Cylinder	C	3	C E A D B F
	Parall square base no PD	D		
	Parall triangular base no PD	E		
	Cylinder no PD	F		
				1 2 3 4 5 6 7 8 9 10 11 12
2	Small No Audio Rep 1	A	1	A I F G C D R M L O P J
	Medium No Audio Rep 1	B	2	R M L O P J I A F C G D
	Big No Audio Rep 1	C	3	A I F G C D L P J O M R
	Small Audio 2D Rep 1	D		
	Medium Audio 2D Rep 1	E		
	Big Audio 2D Rep 1	F		
	Small Audio 3D Rep 1	G		
	Medium Audio 3D Rep 1	H		
	Big Audio 3D Rep 1	I		
	Small No Audio Rep 2	J		
	Medium No Audio Rep 2	K		
	Big No Audio Rep 2	L		
	Small Audio 2D Rep 2	M		
	Medium Audio 2D Rep 2	N		
	Big Audio 2D Rep 2	O		
	Small Audio 3D Rep 2	P		
	Medium Audio 3D Rep 2	Q		
	Big Audio 3D Rep 2	R		

Figure 19: Multimodal Latin Square Order. Trial order is shown for each of the 3 possible latin square order. Note that "medium" size on expo is not used.

Latin square in psychoacoustic

It is better to present the sounds grouped by HRTFs with pauses between each block. The pauses insert a break when the HRTF is going to change. Since all sounds in the same block belongs to the same HRTF, they (their positions) are presented at a total random order. However groups are not presented at a random order but they exploit the Latin Square approach. There is a pletora of articles about latin squares in experiment design and setup, see as a reference Mandl [40]. On a latin square we consider the subjects as one dimension of the square and the blocks as the other dimension. The only effective limitation to this approach is that the number of tested blocks must be the same for all subjects but this is not a problem in our experiment. This approach in used in different disciplines in order to differentiate the effects of previous experiments on different subjects.

For example, suppose that there are 3 new medicines to be tested on different subjects. These medicines could have different effects on a subjects according to the type of previous medicines given to that subject. In order to test these medicines with all possible interactions they are given to the subjects in a different non random order. At the same manner, previous heard HRTF groups could modify the response to the successive groups since there is a strong learning curve and increasing fatigue during the experiment.

As an example consider the situation of 3 HRTF blocks to be repeated 2 times on the same subject (for a total of 6 groups). If we call the 3 blocks A, B and C we will present the blocks in the order reported on table 3:

Subject	Latin Square Order (first repetition)	Latin Square Order (second repetition)
1	ABC	BCA
2	BCA	CAB
3	CAB	ABC

Table 3: Latin square psychoacoustic block order. Each block corresponds to a HRTF.

For the fourth and successive subjects (with ID number n), the experiment is performed with Latin square order corresponding to the one used by the subject with ID $n \, mod \, 3$. The subject is not aware at all of the Latin Square order. He or she also does not known that the stimulus are grouped by HRTF. Also note that two blocks containing the same HRTF are never presented one near each other. Pauses are introduced in order to further reduce fatigue and to attenuate this side effect.

Latin square in multimodal

A similar approach is followed for the multimodal experiments with the objective to give the same positive effects. Each subject is associated with a latin square number (1, 2 or 3) exactly as in the psychoacoustic experiment. Then figure 19 shows how trials are presented to a particular subject according to the latin square order.

4 IMPLEMENTATION AND COMPONENTS

Chapter overview

Our experiments are based on different software and exploit different hardware interfaces. In order to understand our work, real time constraints and tools, and in order to be able to set up a similar environment, it looks useful to analyze the environment that we used and developed. The chapter begins showing software environments (*Matlab* and *Pure Data*) and the communication protocol between them. It also shows the main features of the audio engine. Then it moves on giving an overview of the hardware components and consequently real time constraints. The third section of this chapter explains pinna image acquisition and tracing. The chapters concludes with an overview of the main algorithms/code portions that have been implemented.

4.1 SOFTWARE AND ENVIRONMENT

4.1.1 Matlab

Matlab advantages

Our test system is based on Matlab that is the core application of the entire environment. For this reason, we decided to base the work on Windows™7 that currently offers the best support for Matlab. Because of its features, the famous *Mathworks* software appeared as a natural solution for us, in particular:

- It allows fast creation of graphic interfaces both with programmatic and GUIDE approach. GUIDE is the Matlab solution for the visual creation of interfaces and it is perfectly integrated with the Matlab IDE. It saves us tons of time since the main alternative was to base the entire system on C++ libraries and graphic interfaces on QT;
- It provides functions to easily analyzes data;
- It has powerful computational libraries with a pletora of toolbox for statistics, image manipulation and audio;
- It is a standard in data processing and plotting, especially in the academic world.

Matlab weak point

On the other hand Matlab generally shows poor performances on intensive computing operations since it is Java-based and an interpreted language. However this is not a problem for these experiments: psychoacoustic experiments have some quite intensive computations but they are all offline i.e. mismatch computation and results analysis. The real time section computes single audio stimuli and send them to Pure Data so there is no overhead.

Multimodal experiments are more intensive because they have to compute in real time the TAMO device position, audio source and update the virtual map exploration path. We had to test the performance of this system as described in 4.2.2. The generation of virtual spatial maps (that is sometimes quite intensive) is generated offline when required.

Our experiment system has been tested on Matlab 2010, 2011 and 2012 and it works without any problems. We also tested it on Matlab 2007 and it does not properly works.

Psychoacoustic management

Psychoacoustic Matlab workflow

Psychoacoustic environment in Matlab is organized as follows: the main application is launched by `startMain.m` that calls `subject.m`. This script, linked to a `subject.fig` GUIDE figure displays the database of all subjects and provides callbacks for all functions: database editing, adding new subjects, loading images, image manipulation, image tracing, experiment parameter setup, results viewing and results plotting. Some of these callbacks are just functions, while others start new GUIDE figure, such as image tracing and `experiment.m`. This file is run during the experiment and it completes the following operations:

1. Read input parameters and data from `subject.m`, in particular frequencies values extracted from the pinna image;
2. Use the frequencies values to calculate weighted mismatches with CIPIC subjects;
3. Choose CIPIC subjects according to desired parameters. Actually we choose KEMAR (fixed subject) and the smallest mismatches according to criteria 2 and 3 (see 3.2.2) avoiding duplicates. The algorithm used is described in 4.3;
4. Compute stimuli order (this could be totally random or latin square using blocks. In the latter case, there is the possibility to run only one half of the experiment - the first or the second);
5. Prepare subject's interface;
6. Load Pure Data and load HRTFs in the central memory;
7. Run the experiment, managing pauses;
8. Save the results. Also partials results are saved after each pause.

Psychoacoustic results analysis in Matlab

Data are sent between GUIs using the `setappdata(..)`and `getappdata(..)` functions. Results are always stored as `.mat` files in the *results* directory. Results can both be showed on Matlab console or analyzed and auto-plotted by `parseResults.m`. This script computes:

- front/back confusions;
- updown confusions;
- average response time;
- externalization;

- mean azimuth error;
- mean elevation error;
- mean sphere error;
- linear regression values.

And it plots results comparing the 3 criteria. There is also another file called `parseResultsTwo.m` that calculates the same value calculated by the previous script (except those regarding azimuth and sphere) and split the results by azimuth ranges: front, left, back and right. There is also the possibility to plot the results of the first half of the experiment or just the second half. Results can also be plotted enabling and disabling up/down confusion correction.

Multimodal management

Multimodal Matlab workflow

Multimodal environment in Matlab is organized as follows: the main application is launched by `startMain.m` that calls `subject.m`. Even if this is a different script from the one described in the previous paragraph, it has some common features and it has been implemented in order to be fully compatible with the database of psychoacoustic experiments so there is the possibility to share a common database of subjects. This script, linked to a `subject.fig` GUIDE figure displays the database of all subjects and provides callbacks for all the functions: database editing, adding new subjects, experiment parameter setup and results viewing. This main interface provides the access to the 3 multimodal experiments (experiment 0, 1 and 2). Once an experiment has been selected and launched, the experimenter can choose the map to use. Finally `startMapFigure.m` is launched. This file is run during the experiment and it completes the following operations:

1. Read input parameters for the experiment;
2. Start Pure Data for audio computation (when needed);
3. Create or load the required map;
4. Start a *refresher* that samples the TAMO device position at a configured refresh frequency;
5. Send elevation values to the TAMO device using a COM port and wireless device;
6. Use the sampled position to compute sound marker position, distance and send the data to Pure Data. Also use the position to update the exploration path;
7. Check the exit conditions (such as the reaching of a particular point or a key pressing), exit and save the experiment.

Distance considerations

The distance calculated by Matlab is needed to be converted to a sound attenuation level before be sent to Pure Data. In order to achieve this result we used the *Inverse Square Law* to compute the sound level. This is required because we are simulating a spatial map, where the sound level decays with this law. The experimenter can set the minimum sound level for the farthest position. No near field calculation are performed, in particular when the subject's position is really close the sound marker position, the sound level reaches a maximum constant level. The level and the threshold distance from the marker can be tuned by the experimenter. Some works propose a methods for synthesizing near-field HRTF from far-field HRTFs and they should be considered in further developments [56][31].

Multimodal results analysis in Matlab

Results are always stored as `.mat` files in the *results* directory. Results contain the exploration path, the execution time, as other useful data such as the position of the cube on exp 2. They can be graphically shown using `showResult.m`.

4.1.2 Pure Data and OSC packets

Pure Data overview

Pure Data is one of the most famous free software (modified BSD license) visual programming language that allows to easily compute and generate real-time audio. There are a pletora of book about this software, as reference see [17]. The software works linking blocks (patches) between them. Each block has a particular function and it can be connected using its inlets (inputs) and its outlets (output). A common Pure Data program is a graph on nodes (the blocks) connected between them from inlets to outlets.

Pure Data tasks

Pure Data generates train impulses of white noise and it sends them to the audio engine `cw_binaural` that still figures as Pure Data block. The block used to generate the train is `vline~` that accepts as input a message (a list of strings: each string is a number separated from the previous one by a space) containing the instants of raising and decay of each impulse. Pure Data also manages the headphones compensation (see 4.2.1) using patches `hp_equalization` and `hp_selection`. In addition, since the audio engine `cw_binaural` works with azimuth and elevation coordinates in the range [0-360] and since we are using a different and most diffuse coordinate system with azimuth in intervals [0-180], [-180,0] and elevation in interval [-90, 90], we implemented a code portion in Pure Data to convert the data coming from Matlab to the different coordinates system.

OSC protocol

The data are sent between Pure Data and Matlab using *OSC* (Open Sound Control), a modern message passing protocol used in multimodal environments to send data. More information about OSC can be found on the official website or in [62]. In multimodal experiments we send a OSC packet from Matlab to Pure Data each time that the TAMO position is sampled and the relative sound source position is computed. In order to manage the possibly huge amount of data sent, we use the UDP protocol that does not assure reliability but it guarantees better performances than TCP. Psychoacoustic experiment does not have problems since it simply sends sporadic single packets (one for each stimulus) to Pure Data.

Matlab and Pure Data cooperation

It is Matlab that starts Pure Data and open the UDP socket on

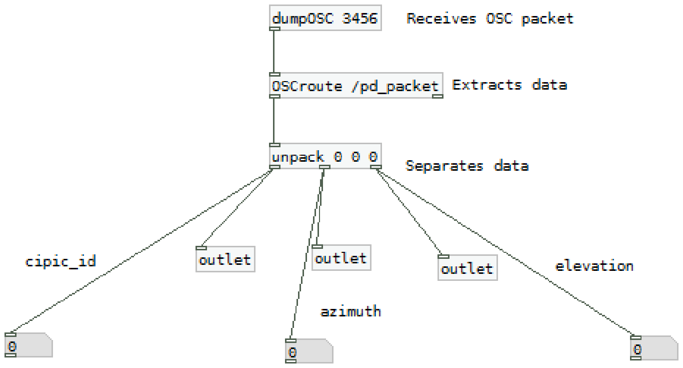


Figure 20: The Pure Data patch that receives OSC packets

localhost:

```
dos('start audio\receiveData.pd');
addpath('../mex');
pnet('closeall');
handles.socket = pnet('udpsocket', 3455);
pnet(handles.socket, 'udpconnect', '127.0.0.1', 3456);
```

The first line starts Pure Data. The second line added required Matlab files used in order to create OSC packets for PureData. The third line closes eventually unproperly managed sockets. Last lines open the new socket on a not-reserved port. Then Matlab can start to send packets to Pure Data with the following syntax:

```
pd_packet = [single(idHrtfCIPIC) single(thetaHrtfCIPIC)
    single(handles.phiHrtfCIPIC)];
[~, datastring]=cstruct(pd_packet);
[~, zerostring]=cstruct(single(0));
string=char(['/pd_packet' zerostring(1:2) ',ffffffff'
    zerostring(1:2) datastring]);
pnet(handles.socket, 'write', string);
pnet(handles.socket, 'writepacket');
```

The Pure Data packet `pd_packet` contains 3 fields: the *ID* of the chosen CIPIC HRTF used to compute the 3D sound, the azimuth (*theta*) and the elevation (*phi*). On multimodal experiments the packets also contain another field: the sound attenuation level. The `single()` function forces single precision data to be sent to Pure Data instead of double values. . Pure Data receives the packets using an ad-hoc patch: it listens to an UPD port with the block `dumpOSC #port`, it extracts data using the block `OSCroute`, then it separates the received values, processes them as described above and it sends them to the audio engine. See figure20

On the first period of pilot experiments, we noted a certain delay (quantified as 10-20 seconds) between the moment when the packets was sent by Matlab and the moment when the sound feedback was received. This only happened on the first packet sent for each CIPIC

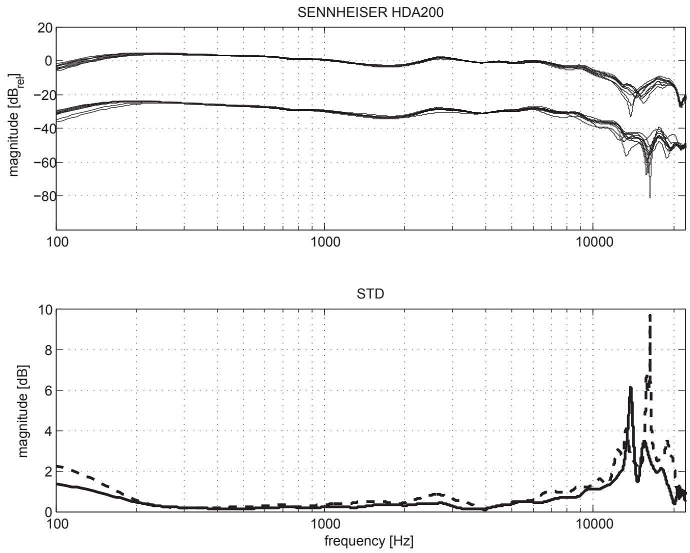


Figure 21: Sennheiser HDA 200 Frequency Response with replacement

HRTF. The problem is that the CIPIC HRTF needs to be loaded into the central memory by Pure Data and this could require some time. In order to avoid this problem, we added a loading phase in which all HRTFs are loaded before that the experiment can start.

4.1.3 Cw binaural engine

CW Binaural overview

The audio engine used in our experiment is the *CW Binaural* engine developed as a Pure Data external called `cw_binaural~`, as described in [19]. It is a powerful engine that supports multiple databases (LISTEN included), in particular it supports the CIPIC database and it offers tools to convert the database to a proper format compatible with the external. For more information see [8]. The main features of this engine are the support of simultaneous different HRTFs, the possibility to choose the interpolation method for intermediate positions (no decomposition or HRTF decomposition in all pass/pure delay in order to better estimate ITD), the possibility to choosen the length of the HRIR considered and the filtering method. At the moment *CW Binaural* does not support distance computation so we implemented this aspect on Matlab and we modulate the audio signal in Pure Data.

CW Binaural parameters

The audio engine is responsible of the interpolation of intermediate positions between the sampled values of the CIPIC database. For more details about the interpolation see [30]. We set up *CW Binaural* to use the following parameters: length of impulse response used for filtering : 128; filtering mode RIFF (it works in the temporal domain and it has no latency but it costs huge computation); interpolating filter Hermite64. *CW Binaural* has been tested in order to achieve real time performances with a 2009 mid-range laptop (CPU Intel Pentium T2330 at 1600 MHz) so in our environment, eve if it has a longer chain, the entire audio synthesis chain has a contained delay of just a few milliseconds as explained in 4.2.2.

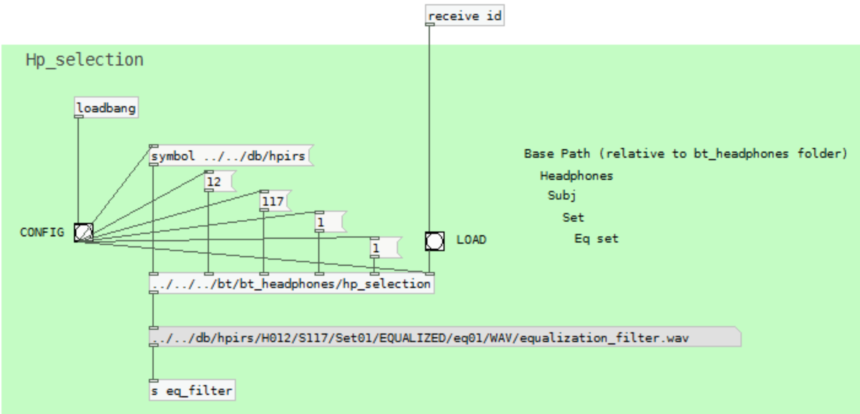


Figure 22: The headphones compensation Pure Data interface.

4.2 HARDWARE

4.2.1 Audio devices

Headphones specifics

In all our experiments we used the Sennheiser HDA 200[1] because they have a very effective passive ambient noise attenuation and they are also able to reproduce very well high frequencies: these dynamic closed circumaural headphones are suitable for audiometry tests as stated by the manufacturer. The frequency response is 20-20000 Hz (see figure 21) with an impedance of 40 Ohm. They are also quite comfortable to be used for a large amount of time, as we need in our psychoacoustic experiment. The only inconvenient that we found using them is that after some time it could happen to hear the beats of your heart. In order to further tune the response of the headphones we added an ad-hoc headphones compensation (see figure 22). This implementation in Pure Data, written by Ing. Michele Geronazzo[2]. Headphones playback usually introduces different types of distortions, in particular the headphones work as a band-pass filter (at 150Hz and 9KHz), modifying the sound frequencies and so highly influencing the 3D sound perception. This effect is particularly visible at frequencies above 7KHz and drastically changes the ILD (Interaural Level Difference): the difference is of about 4dB on 3KHz but is also reaches 12dB on higher frequencies. In order to remove this side effect, Pure Data must compensate these levels, calculated ad-hoc for these headphones. Headphones compensation in these experiments is a common practice, see [51] and [64]. It should also be noted that the Sennheiser HDA 200 frequency response as been calculated on the KEMAR subject without pinna so it is not computed for each single subject of our experiment. This is because in order to calculate personal *HpTF* (Headphones Transfer Function) it is needed to place microphones on the ear and this is quite difficult and expensive on human subjects. Consequently headphones compensation data evaluated for the KEMAR just offer

1 Headphones website: See http://en-us.sennheiser.com/audiometers-high-frequency-testing-closed-dynamic-ear-protector-headphones-hda-200
2 http://www.dei.unipd.it/~geronazzo/

a partial mean compensation on frequencies under 10KHz (for more information see [46]).

External audio card

Headphones are plugged to the system through the external audio card Roland Edirol AudioCapture UA-101[3] that uses an High-Speed USB 2 interface and it is capable of playback 44.1/48/88.2/96/192 kHz.

Silent booth

All experiments have been performed on an silent booth (anechoic room) Sound Station Pro 45 by Puma s.r.l.[4]. It has a normalized acoustic insulation of 40dB in the bandwidth interval of 100-3150 Hz.

4.2.2 Haptic device TAMO

TAMO configuration

The TAMO device introduced in 2.4 is easily configurable with Matlab on Windows systems. It is sufficient to connect both the wi-fi transmitter and the tablet to an USB port and then search for the related COM ported in the system control panel. Drivers are automatically installed. Then the experimenter must insert the COM port on the Matlab configuration script:

```
serial_port = 'COM3';
start_byte = hex2dec('FF');
device_ID = hex2dec('0F');
```

The `start_byte` and `device_ID` are fixed values and they should no be modified. Now the connection with the TAMO device can be opened on the COM port:

```
a = instrfind;
delete(a);
serialPortHandler = serial(serial_port);
fopen(serialPortHandler);
```

The first lines remove eventual previously not properly managed sessions. And finally we can send the packet containing the elevation data for the TAMO lever:

```
data = 90; %lever elevation
data_send = [start_byte device_ID data];
fwrite(serialPortHandler, data_send);
```

The TAMO moves its lever according to the received data and it sends its position to Windows as a common input pen device moving on the tablet. The tablet has the size of the international paper A4 (210mm x 297) even if the area effectively usable by the TAMO is smaller. The lever is moved by a stepper motor that works at 500 hz. The stepper motor is quite noisy and this noise could give important side information to the user, so all experiments (also those without audio)

3 UA-101 website: http://www.roland.com/products/en/UA-101
4 Puma website: http://www.pumasrl.it

have been performed with the circumaural headphones, capable to significantly attenuate the noise. Since TAMO works as a pen device, we sometimes experienced a wiggle noise on the pointer position but it is negligible so we do not add any code in order to compensate this because any additional code would be executed in real time and it could be another significant computational step in an already long processing chain. The elevation data sent to the TAMO depends on the particular TAMO device and should be measured on the physical device in order to convert it to an angular values. This is because the TAMO is still a prototype and different motor could give different elevation values. However data values in the interval 60-130 are often safe values: these values assure a movement range from the ground level (horizontal 0 degress) to about 80 degrees. The TAMO device position is sampled by Matlab on a figure with a refresh rate that can be set up by the experimenter as explained in 4.2.2 and in 4.3.

TAMO final considerations

We tested the TAMO devices on Windows 7 systems and the first time that the w-fi adapter and tablet are plugged, Windows automatically installs the required drivers so no further installation are required. Wi-fi is managed by an *XBee* module with the *ZigBee* protocol[5]. Finally it should be noted that the TAMO devices we used are still prototypes but they already provide a solid and affordable test element.

Stimulus delay

Delay measurement

Since the environment uses different software and hardware interfaces (e.g. COM ports, wi-fi adapters, mouse devices, software interpolation, audio engine and so on) it is critical to verify if it satisfies real time constraints, in particular if multimodal stimuli (audio and haptic) are synchronized within a coherent perceptual integration time window. In order to verify this condition, we measured the delay between the two stimuli. We placed a microphone between the headphones and a microphone near the engine that moves the lever of the TAMO device, then we run an experiment at full load. Since the TAMO's engine is quite noisy it was easy to measure the impulse peak of the movement. We also tracked the corresponding audio train impulse and calculated the time different of the two events as $t_T - t_A$. See figure 23 for an example of recorded tracks. Audio was recorded on different tracks using a Tascam 680 at 192KHz and two condenser microphones. The calculated delay varies according to the refresh rate used in Matlab to extract TAMO's position. We calculated the values listed on table 4.

5 XBee website: http://www.digi.com/xbee/

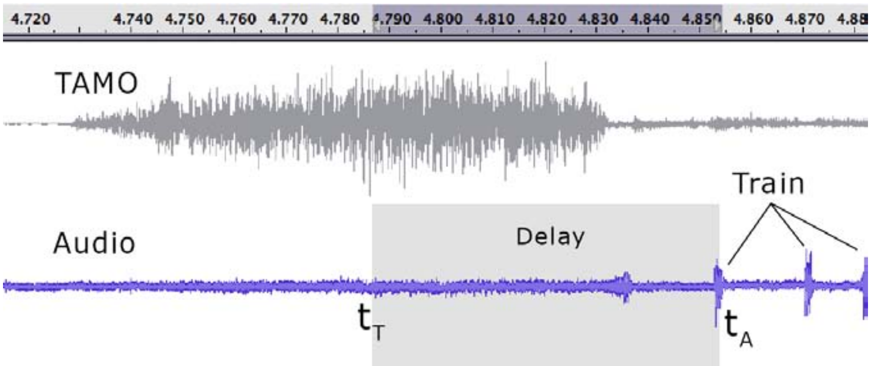


Figure 23: Audio and haptic delay

Refresh rate (s)	Delay (ms)
0.08	68
0.04	53
0.02	61

Table 4: Calculated delays values.

Known real time constraints

Reading the results we can state that a low refresh rate (0.08s) has a 68ms delay that is larger than the delay for a double refresh rate (0.04s). If the refresh rate is too large (0.02s) software and hardware are not able to manage the huge amount of data required and the delay starts to increase again. However all conditions tested provides results that guarantee real time perception for the final subject as reported in [27]: information from different senses occurring at approximately the same time most likely come from the same physical event. Since signals coming from different sensory systems often take different amounts of time to be processed and arrive at a given brain area, there is a relatively wide 'temporal window' of multisensory integration that may last several hundred milliseconds. There are other works confirming the same results, see for example [42] where it is stated that delays of hundreds of milliseconds still enhance the perception, while delays above 200ms start to add a sense of depressions, disorienting the subject. Multisensory interactions can take place over surprisingly long periods of time.

4.3 CODE OVERVIEW

The code that allows the experiments to be executed is mainly written in Matlab. Some code has been visually written in Pure Data and some audio libraries are written in C. Even if the experimental software contains thousands of lines, we would like to show the main algorithms and code portions to allow the reader to better understand the main procedures of our work.

Modular approach

A time-saver modular system

Matlab allowed us to develop the code in a modular way, using functions as `.m` files. The main procedures of our experiments have been defined as functions so it is very easy to add new modules, experiments and so on. Just to give an example, the psychoacoustic database of subjects (`subject.m`) implements a button with a callback that sends configured values of the selected subject to the file `experiment.m`, that is a Matlab GUIDE function. It is immediate to replace this file with a different experiment function or to add a new one: the user does not need to rewrite the entire subject management system and he or she can just focus on the implementation of new experiments. This possibility has already been exploited in our Department on other similar experiments that need to implement features similar to these experiments.

The multimodal map generator is also modular. Multimodal experiments have been tested using a huge number of maps. Each new map can be added in a modular way as a new function. The system automatically recognizes new maps from their name and it correctly handles the pre-running and post-running (saving operations) with the new map.

Finally, the multimodal environment actually supports three experiments called experiments 0, 1 and 2. In order to add new experiment it is sufficient to add a new start button from the GUIDE interface of `subject.m` and to create a configuration `.m` file called `experiment<Number<.m.`, where <*number*> is the number of the new experiment, written as a word e.g. `experimentFour.m`. The configuration file can be easily written copying (and editing as needed) the current content of the existing configuration files e.g. `experimentOne.m`.

Global database and result files

Database overview

The database is a very simple Matlab file called `subjects.mat`. It contains a cell array with 6 columns. The cell array is required in order to properly store string data. The columns are organized as follows:

1. Subject Last Name;
2. Subject First Name;
3. Subject Gender;
4. Subject Age;
5. Subject Pinna Image Path;
6. Subject ID.

The only unique field is the last one (*subject ID*) and it works as primary key for the database. The entire system identifies and sends data about the subject using its ID. It is possible to associate the same pinna to different stored subjects. This feature is particularly useful in order to perform multiple experiments with the same subject: it is sufficient to add a new subject to the database with the same name of the previous subject and then to associate the same pinna image. All contours and traced information are automatically linked, while results files are kept divided. The database is compatible to both psychoacoustic and multimodal experiments developed in this thesis.

Psychoacoustic results format

The subject ID is also used to identify and store results in the `results` directory. In psychoacoustic experiments, this directory contains result files sorted by subject ID. All files that regard the same subject *ID* start with the prefix `ID_`. These files are:

- `ID.mat` that contains all the tested stimulus and subject's answers;
- `ID.pdf` file[6] that contains the main plots of the experiment results;
- `ID_autosave.mat` that contains partial results of the experiment: these results are update at each experiment pause;
- `ID_mismatch_criteria.pdf` files that contain the scores of the CIPIC subjects (mismatches);
- `ID_other.pdf` file that contains other plots results divided by azimuth ranges;
- `ID_part[1-2]` files (both `.mat` and `.pdf`) that contain the results of the first and second half of the experiment.

In order to provide the possibility to generate other plots and to analyze results in different ways it looks useful to analyze the contents of `ID.mat` files. These files contains a matrix where each row represents a single stimulus and the columns are:

1. CIPIC ID;
2. Matched criteria (1, 2 or 3);
3. Chosen azimuth ([-180,0], [0,180]) ;
4. Real azimuth ([-180,0], [0,180]);
5. Chosen elevation ([-90,0], [0,90]);
6. Real elevation ([-90,0], [0,90]);
7. Externalization (1: inside, 2: outside);
8. Distance ([0..1]) from the center on azimuth plot;
9. Useless (backward compatibility);
10. Time to answer in ms (single stimulus);

Multimodal results format

In multimodal experiments, `results` directory contains results files sorted firstly by experiment number and secondly by subject ID. In particular a single result has the following format:

6 Note that you need to generate `.pdf` files by going to edit, parse result in the main GUI.

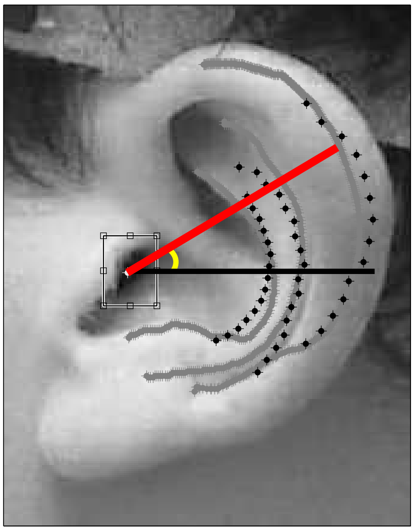


Figure 24: Pinna tracing with angular value. Angle *phi* is shown.

```
exp[num]_sub[ID]_[map]_enablePD[0-1][_rep[0-9]].mat
```

Where `num` is the number of the experiment (currently 0, 1 or 2), `ID` is the subject's ID, `map` is the name of the map, `enablePD` contains 0 (Pure Data disabled - no audio) or 1 (Pure Data enabled) and `rep` eventually contains a number that represents the current trial on multiple experiment on the same map. Each result file contains three values:

- `A` that is a matrix containing the map with elevation values;
- `totalTime` that is the total execution time of the experiment, in ms;
- `explorationPath` that is a *struct* with four fields called `x`, `y`, `soundMarker` and `iterations`.

`x` and `y` are the coordinates of the exploration path on the matrix `A`, `soundMarker` is a vector containing the spatial coordinates of the sound marker on the `A` matrix and `iterations` contains the number of calls to the refresher's callback of the TAMO position.

Contour trace algorithm

Contour trace algorithm extracts frequencies values from the contours of the pinna image in psychoacoustic experiments. It is a core algorithm of this work and it is shown on the following code list:

Listing 1: Contour Trace

```
dist = zeros(3,17);
F0 = zeros(3,17);
for i = 1:3 % cycle contours
  phiIndex = 1;
  for phi = 45:-5.625:-45 % cycle elevations
    minDiff = 360;
```

```
      j = 1;
      while ( Cx(j,i) ~= 0 ) % cycle points
        angle = radtodeg( atan( (Cy(j,i)-earCanal(2)) / (
            Cx(j,i)-earCanal(1)) ) );
        if ( angle >= -45 && angle <= 45 )
          angleDiff = abs(phi - angle);
          if (  angleDiff < 1 )
            if ( angleDiff < minDiff )
              xChosen = Cx(j,i);
              yChosen = Cy(j,i);
              minDiff = angleDiff;
            end
          end
        end
        j = j+1;
      end
      if ( minDiff ~= 360 )
        dist(i,phiIndex) = pdist([earCanal(1), earCanal(2)
            ; xChosen, yChosen]) * pixelToMeterFactor;
        F0(i,phiIndex) = 343.2 / (2*dist(i,phiIndex));
      else
        dist(i,phiIndex) = 0;
        F0(i,phiIndex) = 0;
      end
        phiIndex = phiIndex + 1;
    end
  end
```

It works in the following way: for each contour, and for each angle value *phi* from +45° to -45° at step of 5.626 degrees, find the point `P(xChosen, yChosen)` on the contour that creates with an horizontal line the angle closest to *phi* (see figure 24). Then calculate the distance between the *earCanal* and *P*. Finally convert the distance to frequency using the sound propagation speed and saved it on a vector called `F0`. There is a threshold `thr` value used in order to calculate the closest angle to *phi*: if the angle is larger than `thr`, the point is not considered. If we set a larger `thr` value, more points on the border (near +45° and -45°) are considered, while setting small `thr` value causes the presence of more missing values on the `F0` vector. If no angles are found, value zero is used instead.

In order to find the closest angle, the algorithm scans all points of the contour and keep trace of the current closest match point and its angular value. The distance is scaled using the pixel to meter factor as described in 3.2.

Mismatch values computation

Algorithm analysis

This algorithm stored in the file `calculateWeightedMismatch.m` compares the `F0` frequencies of the CIPIC subjects to the frequencies of the required subjects and computes a rank for the mismatches according to a weighted criteria. Figure 25 shows a plot with a possible output of this algorithm as explained on 2.3.5. The algorithm is used by `experiment.m` in order to select the CIPIC subjects to use in the ex-

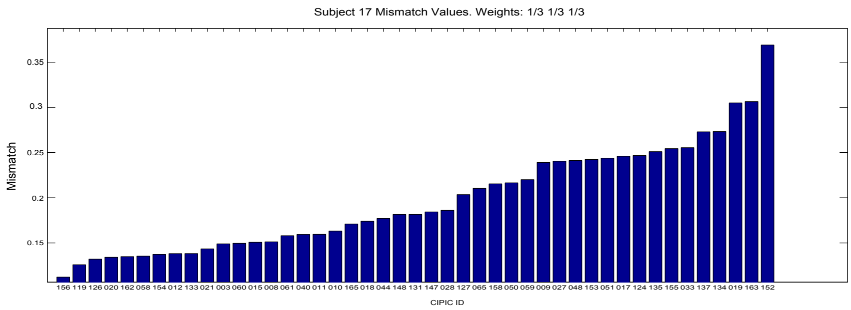


Figure 25: Mismatch values example extracted with the criteria $\frac{1}{3}$, $\frac{1}{3}$, $\frac{1}{3}$

periment. The algorithm has different sub cases and support multiple weights. It works as follows:

```
calculateWeightedMismatch(F0, weight1, weight2, weight3)
  for each CIPIC Subject id
    for each contour
      if empty contour
        mismatch(id, contour) = -1
      else
        mismatch(id, contour) = |F0 - F0_CIPIC(id)|
    end
  end
  for each CIPIC Subject id
    if weight1 or weight2 or weight 3 == 0
      mismatch(id, contour) = -1
    for each contour
      if mismatch(id, contour) != -1
        contourNumber++;
    end
    if contourNumber == 1
      weight(contour not null) = 1
    if contourNumber == 2
      the two contours get half the weight of the third
          contour
    if contorNumber == 3
      use weight1, weight2 and weight3
    sumMismatch(id) = mismatch(id, contour1)*weight1 +
                      mismatch(id, contour2)*weight2 +
                      mismatch(id, contour3)*weight3
  end
  sort sumMismatch
  plot sumMismatch(x=id, y=mismatch)
done
```

In the first section of this algorithm (first *for* loop) we calculate the difference (mismatch) between the current CIPIC subject and our subject. We check both for empty contours and for eventually *null* F0 values (this second checking operation is not reported in the pseudo-code). The second *for* loop calculates the number of available contours and eventually split the weight of a non-existing contour to the other two. If there is only one contour it gets all the weight and the other contours are ignored. Finally a weighted mean is performed and then the values are sorted.

A simplified Matlab version of this code can be found in appendix B.2.

Multimodal map generation

Map generation procedure step by step

Even if we did not perform complete multimodal experiments on a huge number of maps, we tested them as debug, piloting and probing experiments so it looked natural to find out a way to easily generate new maps. Maps are generate by the file `startMapFigure.m`. There is a global variable `A` that is a matrix containing the map. For each x value and for each y value (column and row) there is a z value stored in the matrix that corresponds to the elevation on that point of the map. A map must be configured giving the following parameters:

```
map.xMin = 1;
map.xMax = 640;
map.yMin = 1;
map.yMax = 480;
map.zMin = 60;
map.zMax = 120;
map.soundMarkerX = 320;
map.soundMarkerY = 240;
map.soundMarkerZ = 90;
```

They are self-explained: x, y and z contains the minimum and maximum available values on the map. z values starts from 60 because this is the minimum elevation value supported by the TAMO device (see 4.2.2) and a similar argument is valid for *zMax*. Then the map can be initialized with the following line:

```
A = createEmptyMap(map.xMin, map.xMax, map.yMin, map.yMax,
    map.zMin, map.zMax);
```

There is also a similar function to create an empty map with a perimeter wall:

```
A = createEmptyMapWithWall(map.xMin, map.xMax, map.yMin, map.
    yMax, map.zMin, map.zMax, wallSize);
```

Finally the map is filled with the required figures, for example a cylinder:

```
A = createCylinder(A, map.xMin, map.xMax, map.yMin, map.yMax,
     map.zMin, map.zMax, xCenter, yCenter, height, radius);
```

Where the last four parameters refer to the cylinder. The `createCylinder(..)` function must set the values of the `A` matrix:

```
if (height > zMax - zMin)
  height = zMax - zMin;
end
xDim = xMax - xMin + 1;
yDim = yMax - yMin + 1;
IN = zeros(yDim, xDim);
for y = 1:yDim
  for x = 1:xDim
    if ( pdist([xCenter yCenter; x y]) <= radius )
      IN(y,x) = 1;
    else
      IN(y,x) = 0;
    end
  end
end
A = A .* (~IN);
A = A + IN .* (zMin+height);
```

Some map are CPU-intensive to be generated and the operation could require some seconds on modern PCs, so we decided to save and store the most CPU-intensive maps as `.mat` files in the *maps* directory. In order to add a new map the experimenter needs only to create the function that generates the map `A = create[Name](...)`, to add a new switch case on the function `generateMap(A, inMap, map)` on the same `startMapFigure.m` file and to add the map to an experiment config file `experiment[Number].map`.

Multimodal refresher

Tamo sampling approaches

In order to sample the position of the TAMO device, there are two possibilities:

1. Listen on the Matlab map figure and capture all pointer movements;
2. Sample the position of the pointer with a fixed frequency.

The first approach has the advantage of capturing all pointer movements but it is not guaranteed that Pure Data will be able to process in real time all the packets sent by Matlab: the subject could quicly move the TAMO on the tablet and Matlab will compute and send a huge amount of OSC packets to Pure Data to be processed by the *CW Binaural* engine. Since we need to guarantee real time performances, we preferred the second approach, testing a valid refresh rate (see 4.2.2). We created a Matlab timer that extracts the pointer position at defined instants:

```
refresher = timer('Name', 'Refresher', 'TimerFcn',{
    @refresherCallback, map, socket},
                  'ExecutionMode','fixedRate','Period',
                      refreshRateSeconds);
start(refresher);
```

And once the pointer position has been extracted, its callback computed the position and distance of the sound source and it sends the data to Pure Data. It also extracts the elevation from the map and sends it to the TAMO device:

```
function refresherCallback(~,~, map, socket)
  cp = get(gca,'CurrentPoint');
  id_x = round(cp(1,1));
  id_y = round(cp(1,2));

  % Calculate distance from the sound marker
  dist_2D = norm([id_x id_y] - [map.soundMarkerX map.
      soundMarkerY]);
  intesity_scale_factor = inverseSquareLawScale(dist_3D,
      soundIntensityScale, map);

  % calculate angles
  azimuth = getAngle(id_x, id_y, id_x, id_y - 1 , map.
      soundMarkerX, map.soundMarkerY);
  elevation = radtodeg(atan((map.soundMarkerZ-id_z)/dist_2D))

  % send data to PD
  pd_packet = [single(cipicID) single(azimuth) single(0)
      single(intesity_scale_factor)];

  % send data to TAMO
  data = round(A(id_y,id_x));
  data_send = [start_byte device_ID data];

  % update exploration path
  explorationPath.x(end+1) = id_x;
  explorationPath.y(end+1) = id_y;

  % Check exit conditions
  if ( exitDistance > 0 && dist_2D <= exitDistance )
    saveResults();
    closeExperiment(map.figure);
  end
end
```

The code has been dramatically simplified to be reported here as a reference: do not use it in any real application. The complete callback function can be found on `startMapFigure.m`. A simplified version of this file can be found on appendix C.1.

Helper parsing tools

In order to process the large amount of results, we developed some parsing tools that can automatically extract key data from the results themselves. In particular:

- `parseResults.m` (*psychoacoustic*) automatically merges and extracts data from the first and the second part of the psychoacoustic experiment and plots it as shown on chapter 5. It fixes front back confusions, calculates externalizations, mean errors for azimuth/elevation and linear regressions curvers.

- `parseResultsTwo.m` (*psychoacoustic*) works as `parseResults.m` but it splits the results for azimuth ranges as shown on chapter 5.
- `showResults.m` (*multimodal*) plots virtual maps for all the 3 multimdal experiment (all maps are plotted on the same figure). There is the possibility to add the exploration path and to save the map as *PDF*. There is also the possibility to choose the number of rows and columns in which the subplots shuld be divided, all from a easy GUI-guided procedure.
- `showResultsCubeMerger.m` (*multimodal*) works as `showResults.m` but it also allows to add the position of the cube placed by the subjects on exp2 with the three parameters: x that this the distance from the left margin, y that is the distance from the top margin and size (in pixels).
- `collectDataExpTwo` (*multimodal*) allows to easily process in batch multiple results from exp2 and to associate them with a particular latin square order. Results coming from this script are ready to be imported in a spreadsheet.

5 RESULTS

Chapter overview

This chapter presents the quantitative and qualitative results of our work. On section 5.1 we present the results and plots regarding our psychoacoustic experiment, including some preliminary observations. We also analyze step by step and emphasize some points of particular interest. On section 5.2 we present the results of multimodal experiments, with some preliminary considerations. It should be noted that these results are mainly qualitative and they work as a reference and as a start point for further developments. They are also a background for possible practical application of the HRTF selection developed in the first step of this work. The results are divided in groups by experiment number.

5.1 PSYCHOACOUSTIC EXPERIMENT RESULTS

General preliminary considerations

Before showing the results of our experiments, we would like to present some considerations emerged during the analysis and the afterthoughts on the results.

First of all, we reverted the front/back confusion on azimuth values, fixing them. It has been proven that using non individualized HRTFs may increase front-back confusion, as well as non externalization of the sound sources [19] and fixing front/back confusions is a common and accepted practice. All azimuth values placed in the back hemisphere and that should be on the front hemisphere, are mapped with an axial symmetry to the front hemisphere before calculating azimuth and elevation errors, and vice-versa. We also check (as a reference) but do not correct the number of up/down confusions that is the number of stimuli perceived above the eye-level and that was under and vice-versa. We noted that front back confusions are larger with generic HRTF (KEMAR) as confirmed in [32].

It should also be noted that a not negligible amount of azimuth and elevation error is caused by the HRTF interpolation (see chapter 2). *CW Binaural* engine reports a possible error of 5 degrees in azimuth and 5.6 degrees on elevation. We do not mathematically take into account these values in the final results.

We also tested the results of completely random elevation values inserted by a random number generator in the interval [-90 +90] degrees, with uniform linear distribution. The mean error converges to 45 degrees so subjects with elevation error of about 45 degrees are not reliable. We found only 1 subject in the first half of the experiment with these large elevation error values. We generally saw increase performances in localization between the first repetition and the second one proving that maybe future experiments should consider a longer training sessions.

It is renown that personal HRTFs influences mainly elevation, in fact we was a large improvement on elevation but no improvement on azimuth localization tasks using selected HRTF instead of KEMAR.

We also performed short informal interviews to the subjects asking them their opinions about the experiments. One of the most interesting response is that a significant number of subjects perceived a large part of the sounds coming from the back, in accordance with the experiment results.

Analyzing the results, it is also clear that externalization is a really subjective perception: it works very well for some and it works very bad for others. There could be other factors that can improve externalization such as reverberation, individualized headphones compensation and ear canal compensation.

Results of previous works

Previous works provide interesting results to be reported here as a reference in the successive analysis. Some of these works have already been mentioned in 2.5.

Blauert et al. [7] found localization blur on the horizontal plane between 3 degrees (front sources), 10 degrees for lateral sources (probably due to the cone of confusion), and 5 degrees back sources. On the median plane, the localization blur calculated reached values up to 22 degrees. This is not a localization error but a localization blur: the minimum angle difference in order to identify a different source location.

Douglas et al. [18] affirm that with their system they can reduce the elevation error as much as 33 degrees compared to KEMAR.

Zotkin et al. [64]talk about an increase performance of 20-30% between a generic HRTF and a chosen HRTF with biometric parameters. They used only 6 subjects and placed all sound sources ahead with elevation in interval [-45, +45], also they only presented 20 stimuli. They used head tracking but no headphones compensation . As input they used an hand pointer so it is impossible (or quite unnatural) for such a system to test back position on the horizontal plane. In our experimenter the choice is completely free. We also allow to choose an elevation point on the range from -45 degrees to -90 degrees where effectively there are no sounds presented.

Begault et al [5] calculated a mean error from 18 to 25 degrees on azimuth and reversal rate from 28% (head tracking) to 60% (not head tracking). Externalization on anechoic condition was of 40%. Their absolute elevation error is comparable to the values that we are going to present.

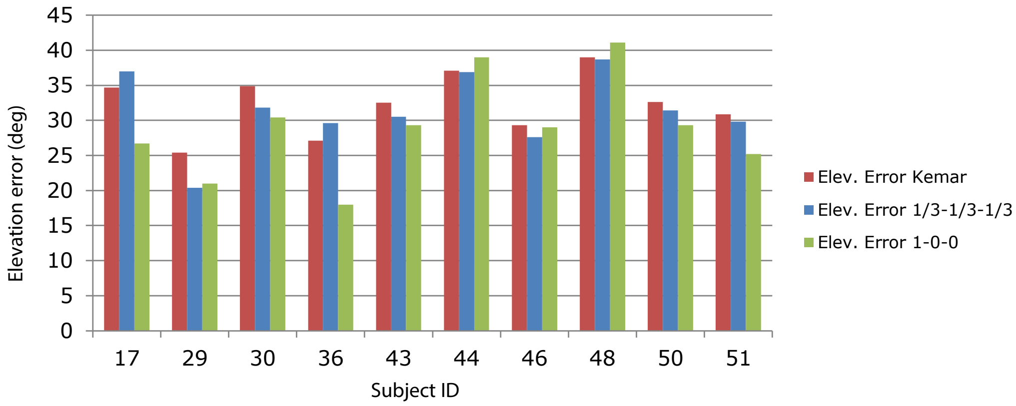


Figure 26: Psychoacoustic Results. The criteria $\frac{1}{3}$, $\frac{1}{3}$, $\frac{1}{3}$ is definitely the best one.

5.1.1 Results overview

ID	Elev. err. (KEMAR)	Elev. err. (1/3)	Elev. err. (100)	Improvement Kem./100	Lat. Sq.
17	34.7	37	26.7	23.1%	3
29	25.4	20.4	21	17.3%	1
30	34.9	31.8	30.4	12.9%	2
36	27.1	29.6	18	33.6%	1
43	32.5	30.5	29.3	9.8%	1
44	37.1	36.9	39	-5.1%	2
46	29.3	27.6	29	1.0%	3
48	39	38.7	41.1	-5.4%	2
50	*32.6*	*31.4*	*29.3*	*10.1%*	-
51	*30.88*	*29.8*	*25.2*	*18.4%*	-

Table 5: Psychoacoustic Summary Results. Errors are expressed in degrees. Subject 50 is a fictitious subject that contains the values of all the other subjects so it contains mean values of all the previous subjects. Subject 51 is defined as subject 50 but it does not contain the two outliers with IDs 44 and 48. Standard deviation of the error is 25.3 degrees.

Criteria (1, 0, 0)

Table 5 contains a summary of the results of the entire experiment. The next pages will show accurate and extensive results for each subject, showing both azimuth and elevation values. They will present the results both in a global form and split by azimuth areas. Since vertical localization is the most relevant task for the pinna reflection cues (the model that we used in the HRTF selection) this short table just report the elevation error between the simulated audio 3D positions and positions chosen by the subject. Figures 26 and 27 are two plots of the results. These are divided by HRTF according to criteria 1 (KEMAR), criteria 2 (weigths $\frac{1}{3}\,\frac{1}{3}\,\frac{1}{3}$) and criteria 3 (weights 1, 0, 0). It is clear that the reference KEMAR has the average worst performance. Criteria (1, 0, 0) scores the best results and it could be considered the best criteria at the moment. It gives an average improvement on the performances of 10% with a peak of 33.6% compared to the KEMAR, confirming that the most external contours (C2/C1: helix inner wall and helix border)

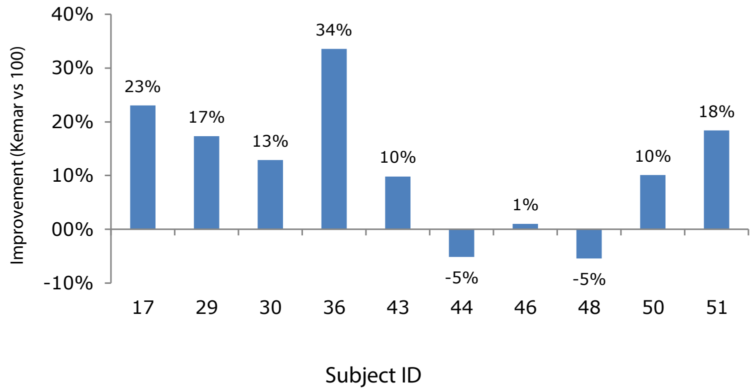


Figure 27: Psychoacoustic Increment (percent) in results comparing the KEMAR with the criteria $\frac{1}{3}$, $\frac{1}{3}$, $\frac{1}{3}$.

are the most significant and information-rich for elevation cues with this setup (generic headphones compensation and using this selection criteria). There are 2 outliers who slightly decrease their performances of 5% with this criteria. One of these subjects, however, gave results comparable to total random answers in the first half of the experiment. Furthermore both subjects increased their performance in the second half of the experiment. Subjects 51 contains the global results without these outliers. It is quite curious to be observed and it surely requires more studies the fact that the 2 outliers are the 2 female subjects. It should also be noted that with criteria (1, 0, 0) the KEMAR is a good chosen HRTF for both subjects (it has a mismatch smaller than the one computed with half of the remaining CIPIC subjects).

Criteria $\frac{1}{3}\,\frac{1}{3}\,\frac{1}{3}$

At a quick look, criteria $\frac{1}{3}\,\frac{1}{3}\,\frac{1}{3}$ seems unreliable since it sometimes offers better performances than KEMAR while sometimes it shows poor performances. This could be related to the *non-individua*l headphones compensation. On the other hand, a deeper analysis shows that the poor performances compared to the KEMAR are smaller (-6% subject ID 46 and 48) than the better performances (+39%, subject ID 36) compared to the KEMAR. Additionally the 2 worst poor performances are achieved by the 2 outliers and the best performance is greater than the best performance of the criteria (1, 0, 0). In conclusion this criteria offers a huge amount of possible improvements but it is actually not enough safe and tuned. Maybe one of the biggest problem is the fact that headphones are not individual-compensated and this could cause problem above 10kHz, where the two most internal contours are more effective. Furthermore another possibility is that the weight of the two internal contours is too large compared to the weight of the most external one. An interesting point is that it is more common to insert errors in the tracing process of the most internal contours since they are closer to the focus point and therefore the distances are smaller introducing a possible noise in the selection criteria. Future developments proposed on chapter 6 contain a possible solution that consist in choosing the HRTF according to the 3 criteria in different steps.

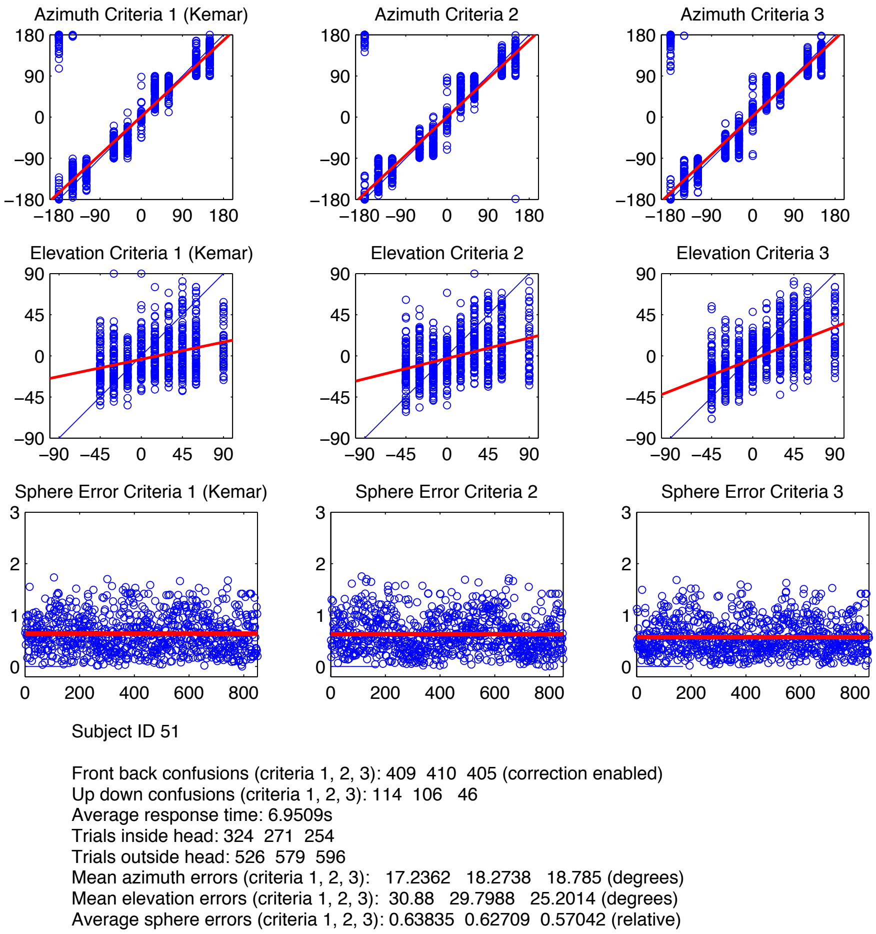
Latin square

It is also significant to compare the results considering latin square order. The two outliers performed the experiment with latin square order number 2, that presents the criteria (1, 0, 0) on blocks 2 of 6 and 4 of 6. This latin square presents the criteria earlier compared to the other latin square orders. Since we noted a learning curve during the experiment (as explained in the following of this chapter) for these 2 subjects, the effects on criteria (1, 0, 0) were smaller.

5.1.2 Global results

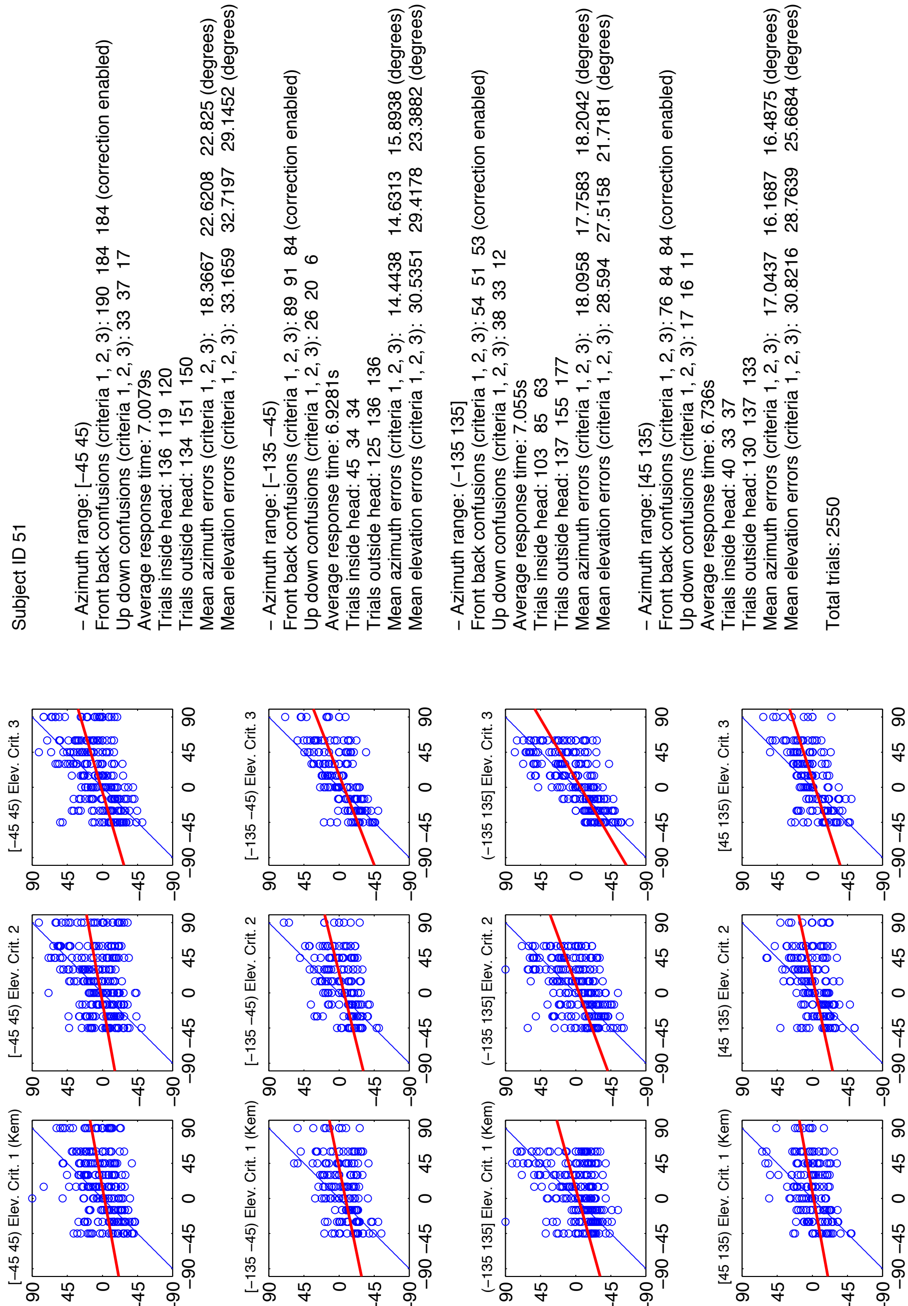
In this paragraph we analyze detailed global results for all the subjects except the 2 outliers. However, as we already said, these 2 outliers show an increasing trend between the first and second half of the experiment.

Figure with Subject ID 51 reports the global results. In the first row there are the azimuth results divided by criteria (KEMAR and selected HRTF). These plots should be read in a circular vertical way e.g. azimuth of about 180 degrees is closed to -180 degrees. In the second row there are the elevation results divided by criteria whom results have already been discussed in the previous paragraph. The last row of plots is calculated in the following way: each tested 3D point is mapped on a sphere and at the same manner it is mapped the point chosen by the subject, then the distance vector between these 2 points (error) is calculated and plotted. This third criteria combines both azimuth and elevation errors. The red lines report linear regressions. At the bottom there are some useful data summarized.



As we can see, HRTF selection does not modify the performances on front/back confusions and also it does not affect azimuth values. This is totally correct and an expected result because the contours extracted from the pinna only influence elevation cues, as discussed in 2.3.5. On the other hand it dramatically decreases the number of up/down confusion e.g. the number of times that a sound coming from above the eye-level is perceived under the eye-level and vice-versa. It also significantly increase the quality of externalization. On both these aspects the criteria (1, 0, 0) is the best one. As previously stated, the criteria (1, 0, 0) increases the average performances on elevation by 18.4% compared to KEMAR. On the new plot of spherical error that consider both azimuth and elevation, the average performances increase of 11%.

Elevation of +90 degrees is the most difficult to be recognizes since the sound comes exactly from above the head. Results further increases removing this angular value. In the results presented here +90 degrees value was not removed.



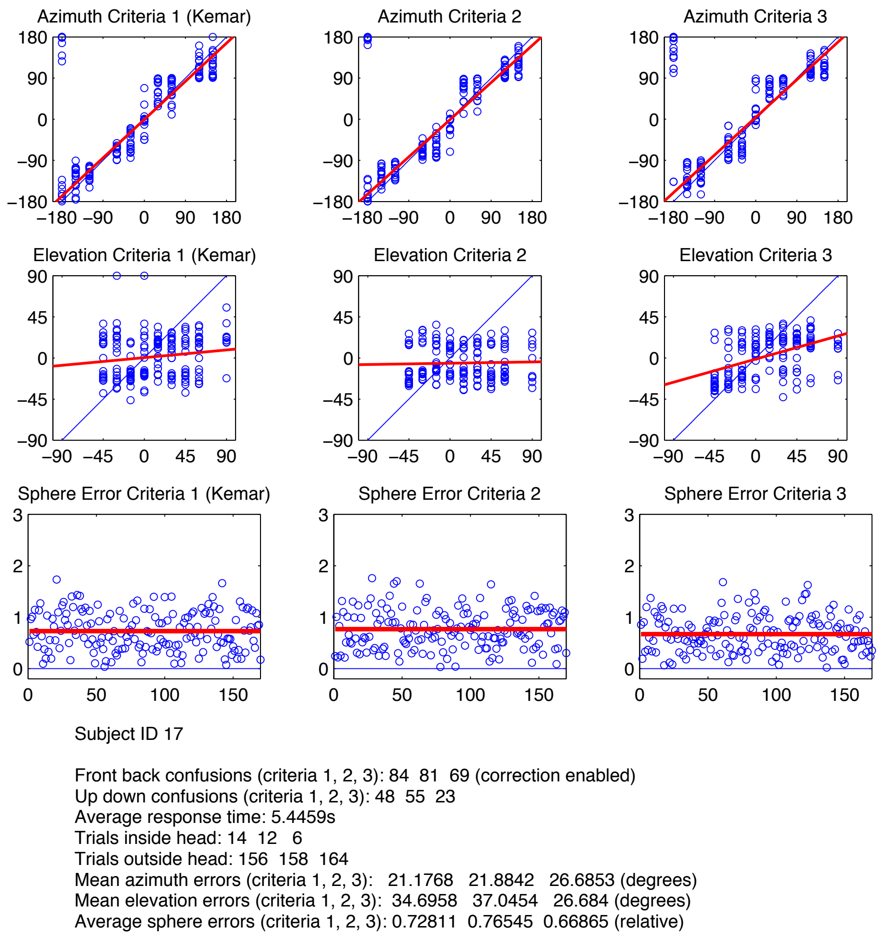
The plots above presents the elevation results (as stated they are the most significant for us) split by azimuth range. We tried to understand how the performances vary according to the azimuthal position of the sound source. The best performances are achieved on the back sector (improvement of 28% compared to KEMAR) in accordance to the fact that we perceive in a better way the sounds backward coming, as stated on chapter 2.3.1. This finding highlights that the HRTFs selection criteria developed in the front median plane (as explained on paragraph 2.3.5) is robust and positively effective also in the posterior listener space. Increased performances compared to KEMAR can also be found on the front section and in the two lateral sections (left and right).

Since we captured images from the left pinna, we compared the results from the left and from the right portion. They are really close so we can affirm that the particular ear chosen in the tracing operation is not significant. With high probability this is because there is a strong symmetry on our head.

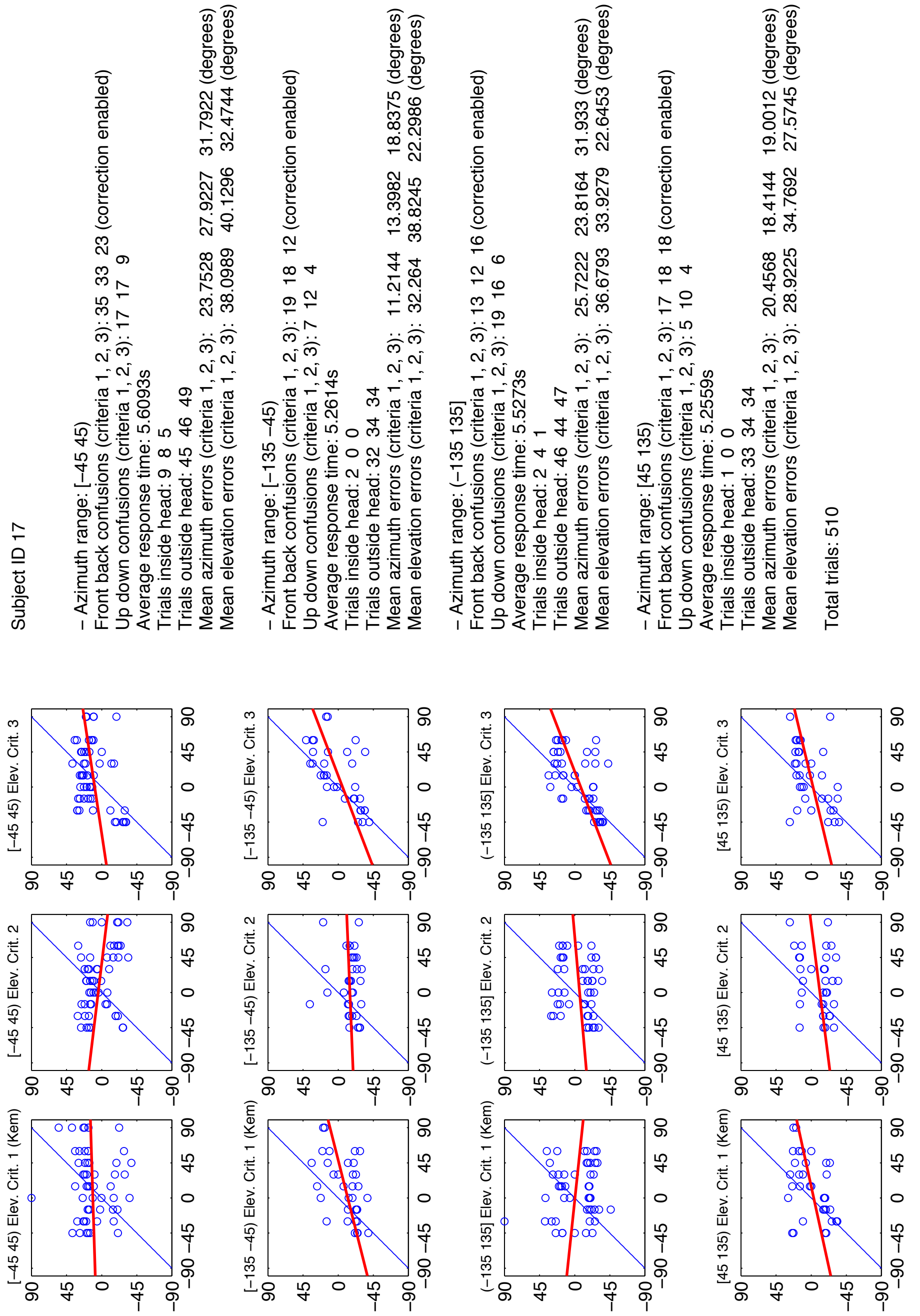
5.1.3 Single Subject's results

Subject 17

This subject increases elevation performances of 23% in respect to KEMAR.



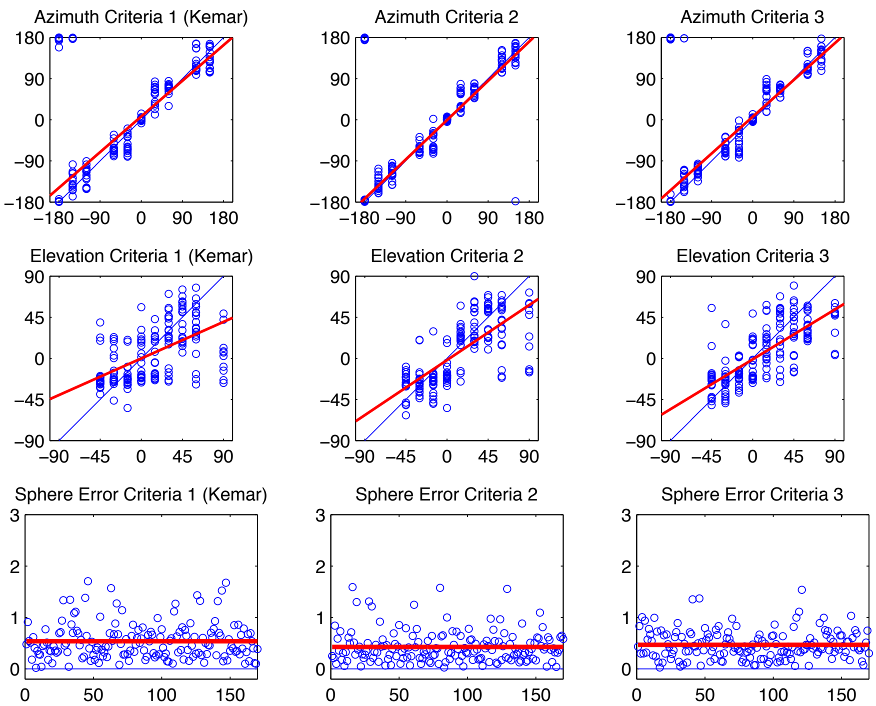
With increased results of 23% in respect to KEMAR, this subject achieved the second largest increment of performances showed by criteria (1, 0, 0). He or she is also the quickest subject of the entire test with an average response time of 5.4s. It is also interesting to note how nearly the totality of sounds are perceived as outside the head by this subject. He or she does not score a good performance with the second criteria ($\frac{1}{3}\,\frac{1}{3}\,\frac{1}{3}$). The linear regression of the elevation for the third criteria is clearly the best one. however, with all criteria a large amount of sounds are perceived in the interval [-45°, +45°] suggesting that the highest elevation are not correctly perceived. There is also quite a large variance around zero degrees. Azimuth errors are larger than other subjects but still valid and coherent.



In the front area the second criteria is definitively the worst, as clearly showed by the linear regression. Best results are in the back area, with an increased performance of 38% using criteria 3 in respect to KEMAR. Surprisingly there is a sort of unsymmetrical situation between the left and the right side, with a difference of about 20% in vertical localization on the third criteria.

Subject 29

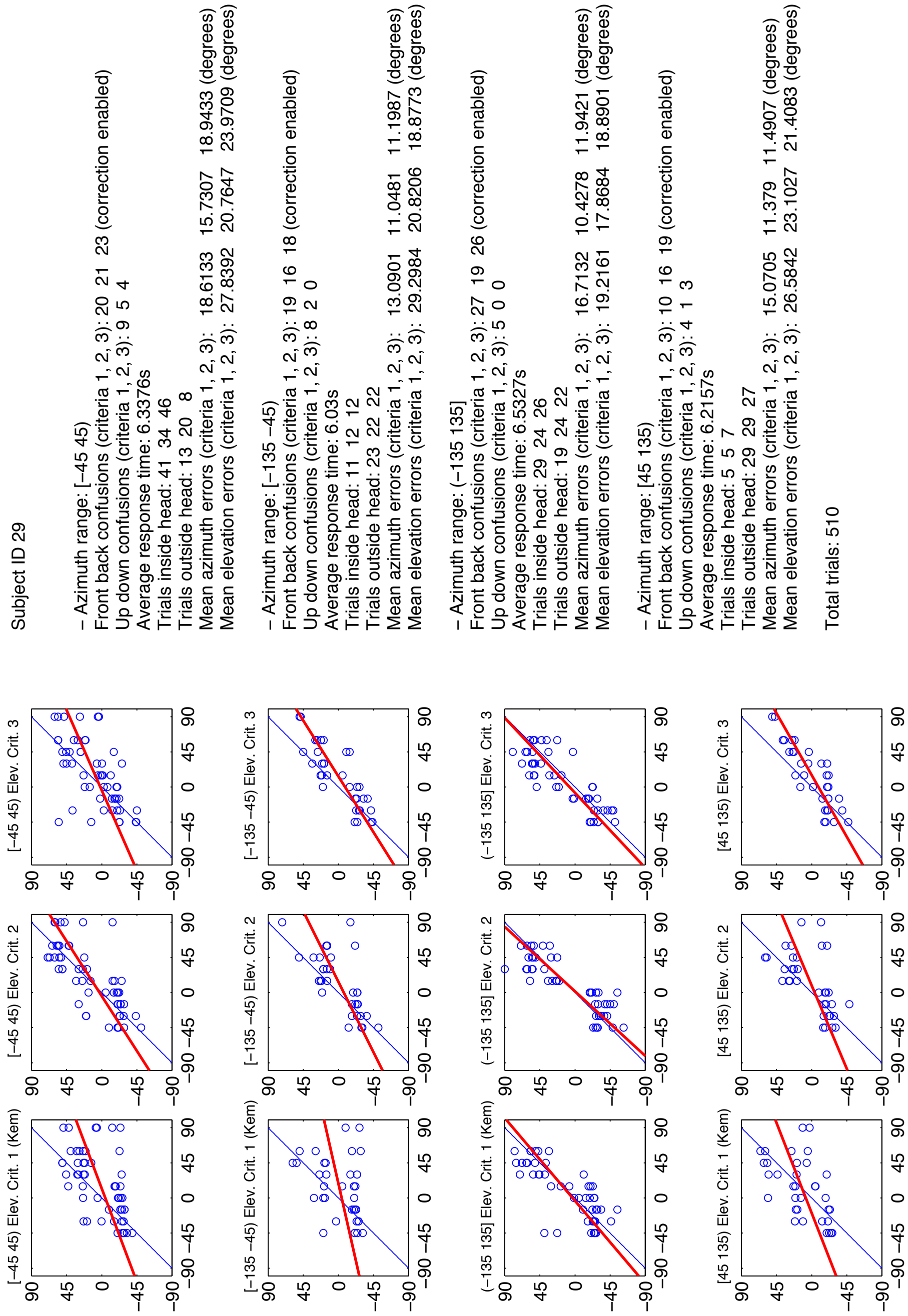
This subject increases elevation performances of 17.3% in respect to KEMAR.



Subject ID 29

Front back confusions (criteria 1, 2, 3): 76 72 86 (correction enabled)
Up down confusions (criteria 1, 2, 3): 26 8 7
Average response time: 6.3068s
Trials inside head: 86 75 91
Trials outside head: 84 95 79
Mean azimuth errors (criteria 1, 2, 3): 16.2301 12.333 13.8035 (degrees)
Mean elevation errors (criteria 1, 2, 3): 25.4453 20.4257 21.0051 (degrees)
Average sphere errors (criteria 1, 2, 3): 0.5357 0.42592 0.46657 (relative)

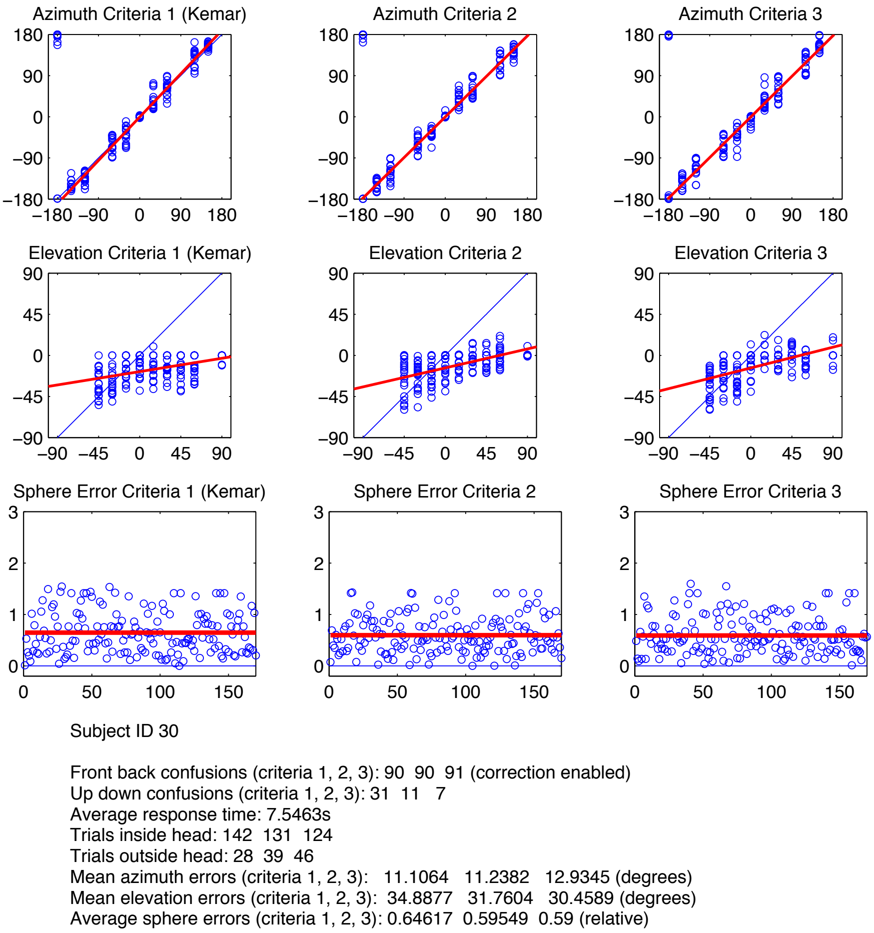
Even if this subject has not one of the largest percent increment, the absolute error values are quite small both for elevation and azimuth: the average azimuth error is of just 14 degrees, while the elevation error with the third criteria is of 21 degrees, just 1 degree more than the second criteria that works pretty well for this subject. The average response time is under the mean value. Externalization seems to do not work very well for this subject: half sounds are perceived as inside the head and the other half as outside the head. The third criteria reduces (and almost removes) up/down confusions. Linear regressions are really close to the real 45 degrees line both for criteria 2 and 3 and both for azimuth and elevation.



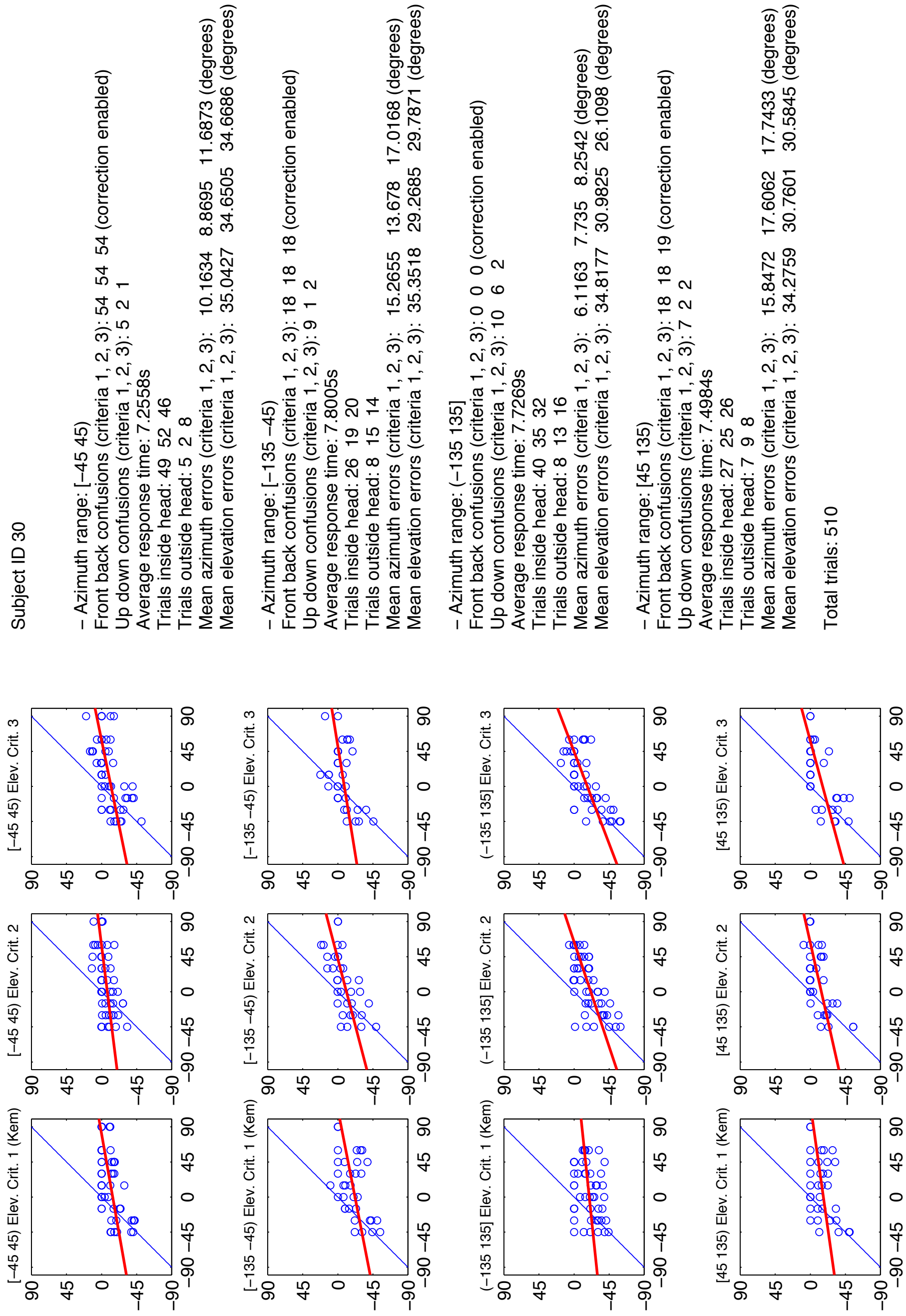
This subject scores almost identical results in the rear area with the three criteria, while there is a significant increment of performances (14%) for the third criteria on the front area, near the median plane, where the elevation cues have been studied. There is also a larger increment of performances on the two side areas, on the left and on the right. This subject is one of the few with the rear area of the third criteria containing a linear regression entirely above the reference 45 degrees line.

Subject 30

This subject increases elevation performances of 12.9% in respect to KEMAR.



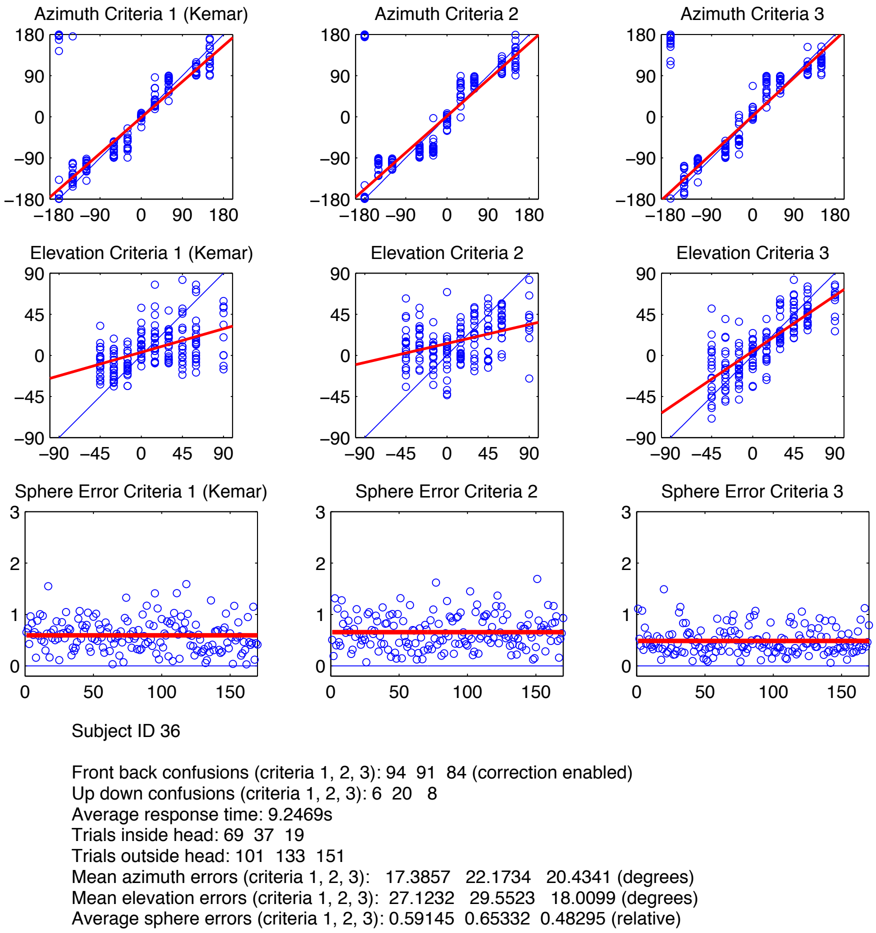
After the experiment completion, the subject told us that he or she heard just a few sounds coming from the front area: he perceived almost all sounds coming from the back area. This is confirmed by the fact that there are about 90 front/back confusions for each criteria for a total 271 front/back confusions that are about half the number of the total stimuli. However the absolute azimuth error of this subject is definitely the smallest of the entire experiment. Externalization does not work very well for this subject and the average response time is really high. As an interesting point, on elevation there is not one single sound perceived above 22 degrees.



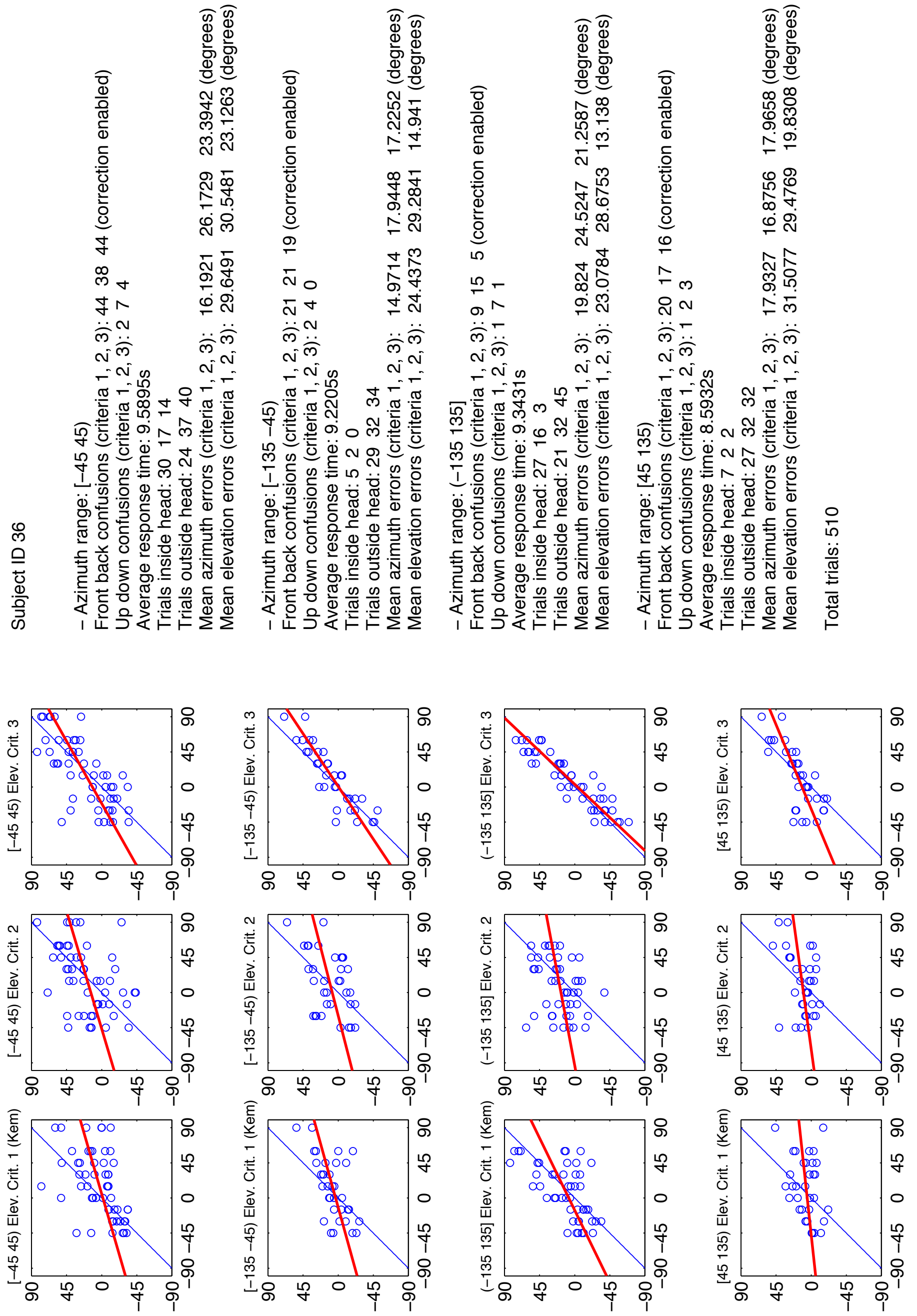
In the front area there is not a significant improvement thanks to HRTF selection. This could be related to the small amount of sound rendered in the front area and perceived coming from the front area. In fact, on the back area, there is a greater improvement (25%) between the KEMAR and the criteria (1, 0, 0). The second criteria seems to perform as (or a bit worse) the third criteria.

Subject 36

This subject increases elevation performances of 33.6% in respect to KEMAR.



This subject shows the larger increment in the performance using the criteria (1, 0, 0) in respect to KEMAR. While front/back confusions are quite large (and comparable with the previous subject), the externalization is working pretty well, in particular with the third criteria. The average response time for this subject is one of the largest, maybe according to the fact that this subject is the eldest taking part at our experiment. The absolute elevation error for the third criteria is one of the smallest (18 degrees). It is interesting to note how, while the third criteria is perfectly working, the second criteria is even worse than the KEMAR. However the HRTF selected with the second criteria is in the 4th position using the third criteria. The KEMAR is still in the first quarter. This is a prove that the mismatch computation cannot be used as a total ordering criteria.

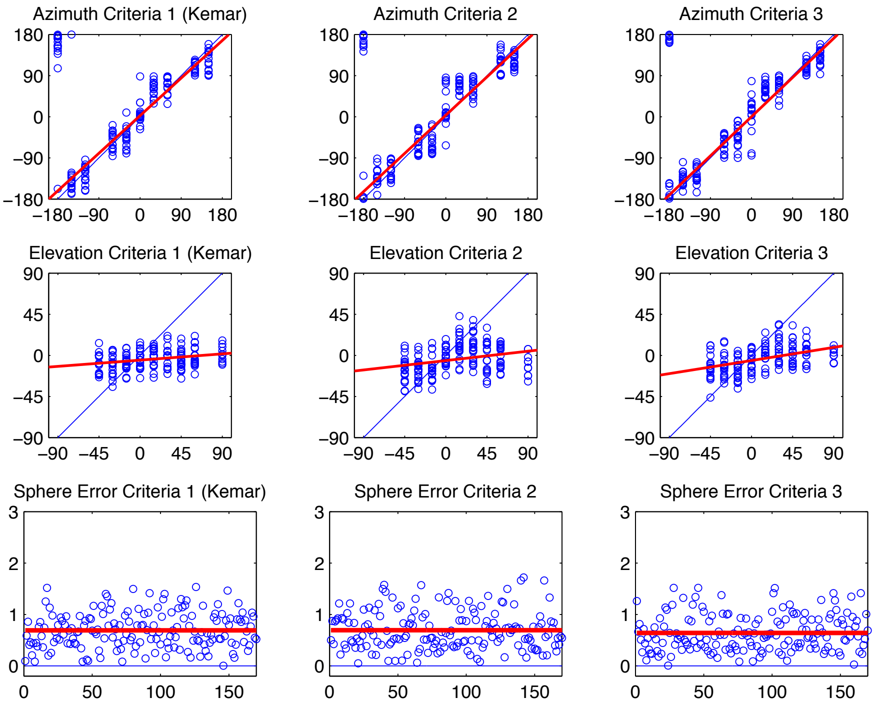


As expected, in each of the four section, the third criteria is the best one. However on this criteria (and only on this) there is a strong asymmetry between the left and the right areas. The improvement in the front area in respect to KEMAR is significant (22%), moreover it is very large in the back area where it reaches 43%. The linear regression of this criteria in the back area is also the only one of the entire experiment with an angular coefficient larger than 1 (larger than 45 degrees with x axis).

The linear regression for the azimuth in the second criteria has negative angular coefficient. The HRTF selection with the third criteria increases the performances in respect to KEMAR more in the front area than in the back area, that is quite interesting because we usually observed an inverse trend.

Subject 43

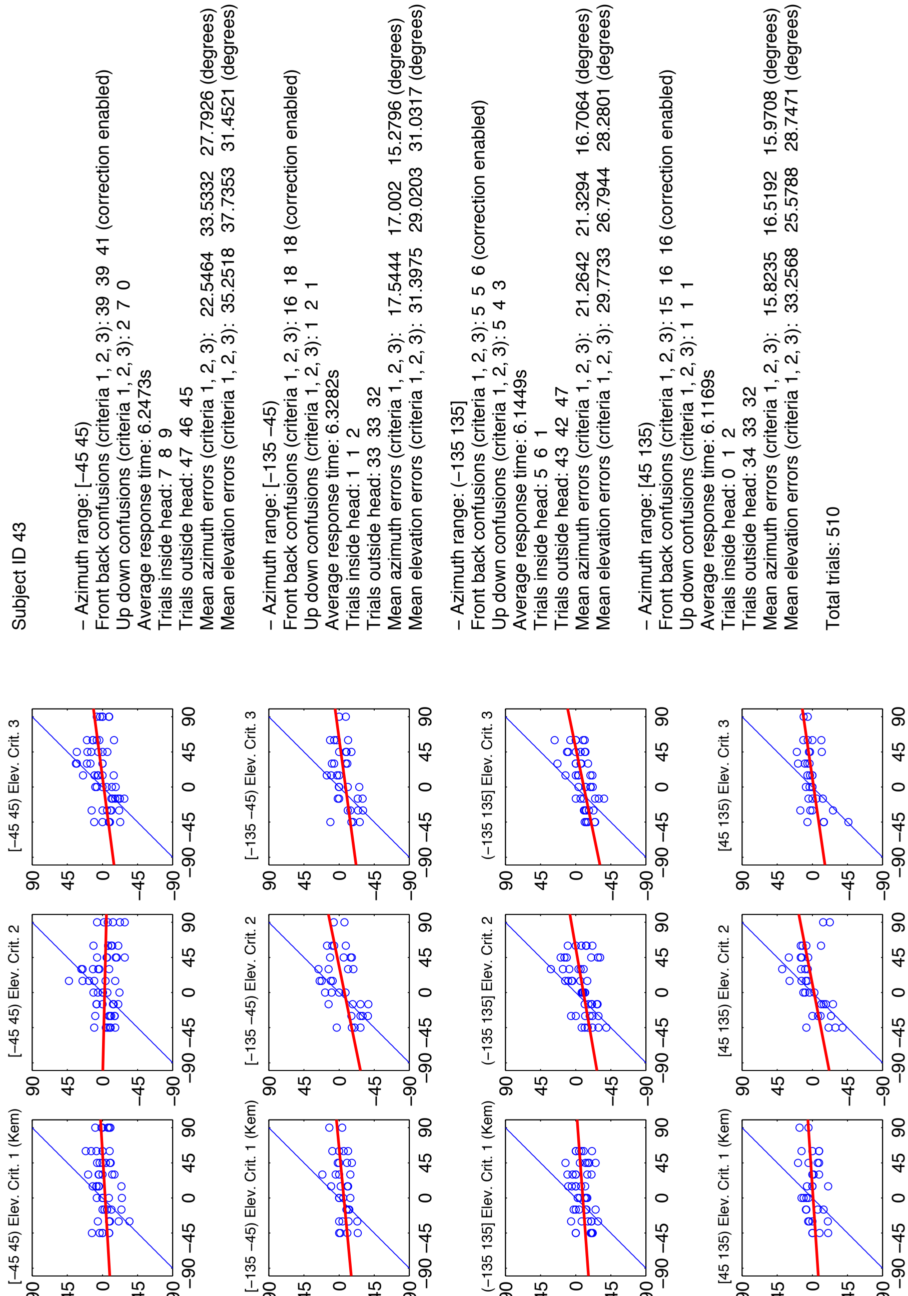
This subject increases elevation performances of 9.8% in respect to KEMAR.



Subject ID 43

Front back confusions (criteria 1, 2, 3): 75 78 81 (correction enabled)
Up down confusions (criteria 1, 2, 3): 9 14 5
Average response time: 6.2085s
Trials inside head: 13 16 14
Trials outside head: 157 154 156
Mean azimuth errors (criteria 1, 2, 3): 19.8168 23.163 19.5998 (degrees)
Mean elevation errors (criteria 1, 2, 3): 32.5351 30.4718 29.9314 (degrees)
Average sphere errors (criteria 1, 2, 3): 0.68889 0.69329 0.63942 (relative)

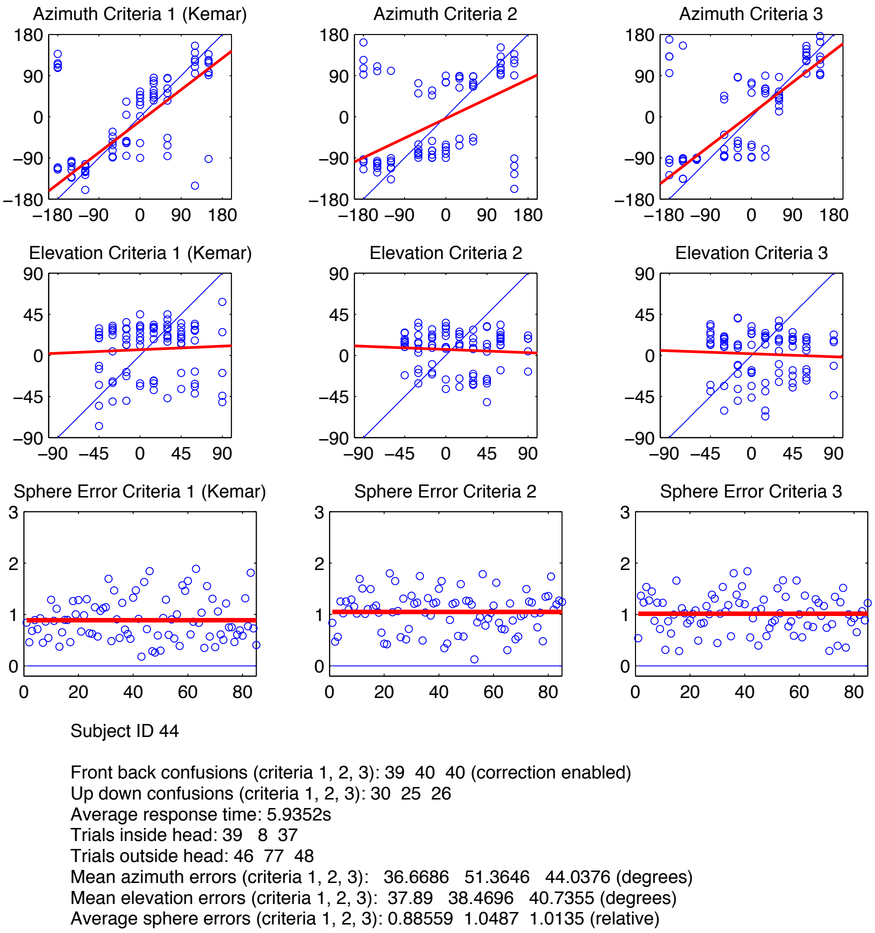
This subject does not show any additional particular key points. Results are positive and they confirm the usefulness of HRTF selection with pinna reflection patterns. Externalization is nearly perfect and the average response time is low. With the sphere criteria (and also only on azimuth, that is not our interest) the second criteria is a bit worse than the KEMAR.



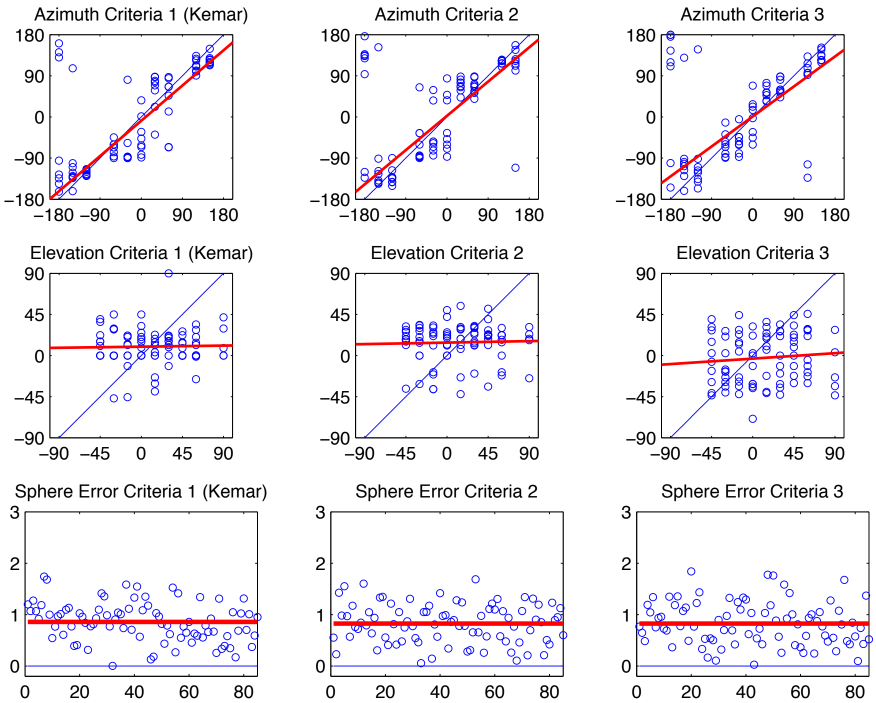
The linear regression in azimuth for the second criteria has negative angular coefficient. The HRTF selection with the third criteria increases the performances in respect to KEMAR more in the front area than in the back area, that is quite interesting because we usually observed an inverse trend.

Subject 44

This subject decreases elevation performances of 5.1% in respect to KEMAR.



Since the results of the entire experiment for this subject are not of great interest, we decided to report here the results of the first half of the experiment (image above) and the results of the second half of the experiment (image below). This is because the results of the first half are nearly random. As stated at the beginning of this chapter, giving random responses provides results comparable to the first half of this experiment. There is an incredible error of 50 degrees in elevation for the second criteria and about 40 degrees of elevation error for the third criteria (near the 45 degrees value of random answers). Keeping these results in mind, the second half of the experiment is really interesting:



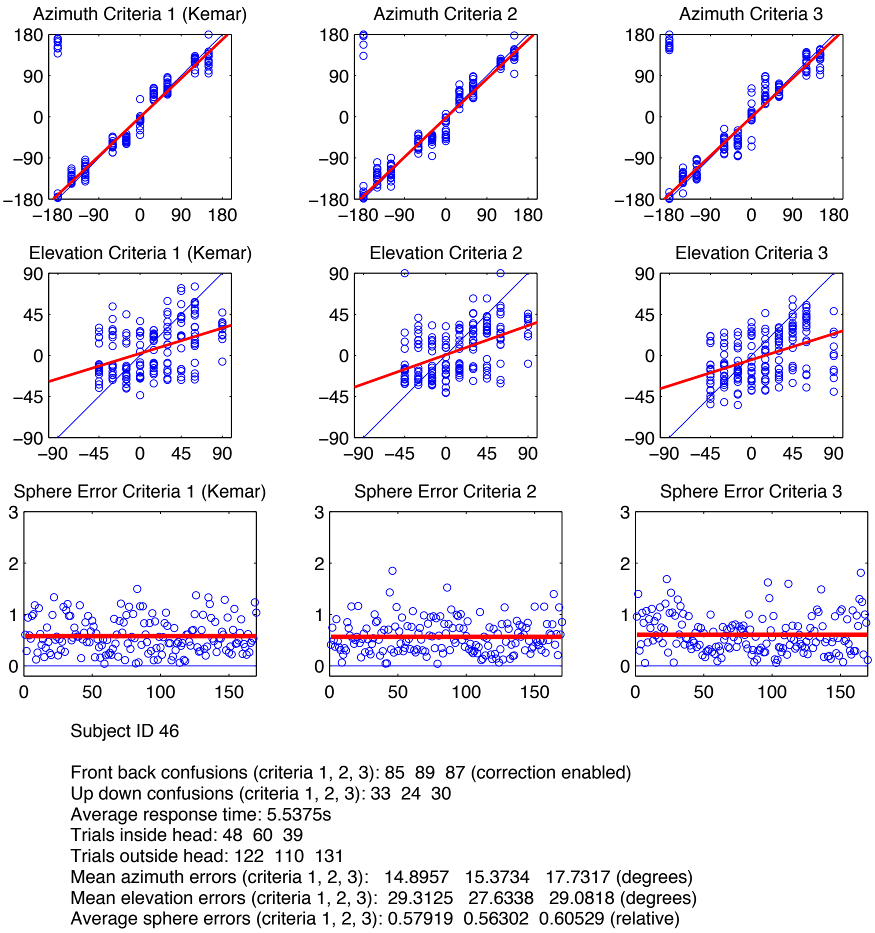
Subject ID 44

Front back confusions (criteria 1, 2, 3): 43 41 45 (correction enabled)
Up down confusions (criteria 1, 2, 3): 16 26 23
Average response time: 5.4666s
Trials inside head: 22 24 34
Trials outside head: 63 61 51
Mean azimuth errors (criteria 1, 2, 3): 34.0942 32.3482 28.5404 (degrees)
Mean elevation errors (criteria 1, 2, 3): 36.3833 35.3203 37.2734 (degrees)
Average sphere errors (criteria 1, 2, 3): 0.85769 0.824 0.82617 (relative)

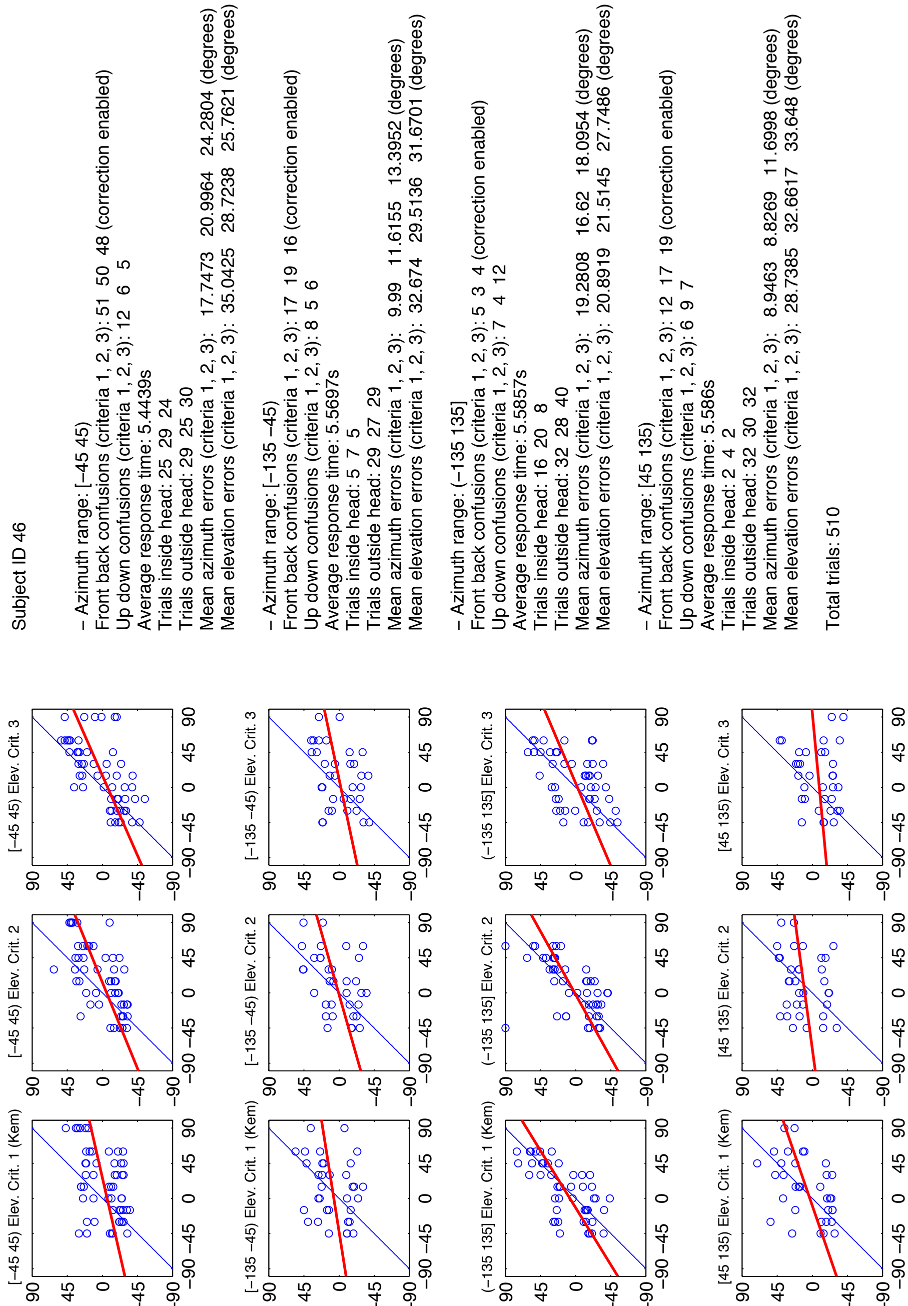
Results exhibit a better trend in the second half of the experiment. This subject was completely naive to 3D audio experiment and probably his or her learning curve is very slow-growing. The fact that some subjects possibly have such a slow learning curve must be carefully considered in future experiments. In the second half of the experiment performances dramatically increased of 40-50%. Even if the three criteria have about all the same error, the absolute value is definitively smaller in this second half. In ordert to obtain a benefit from HRTF selection, further training should be done, and this could certainly be an interesting future development.

Subject 46

This subject increases elevation performances of 1% in respect to KEMAR.



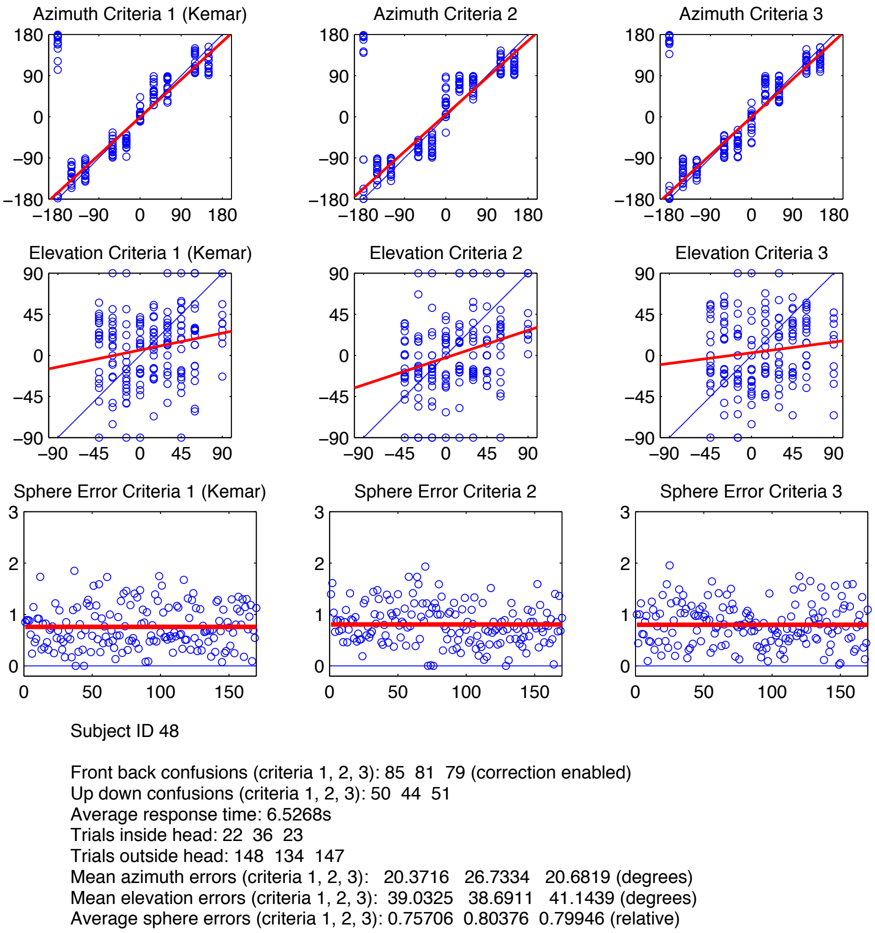
This subject has one of the lowest response time and he or she is also one of the most accurate on azimuth: this is quite particular because we generally noticed that it requires a bit more time in order to precisely place the azimuth. However he or she is also the youngest subject taken part at our experiment so the response time could be related to this aspect, as the eldest subject had the higher response time. Externalization works very well for this subject but there is not a significant improvement on performances with the third criteria. The second criteria has a better increment of the 6%.



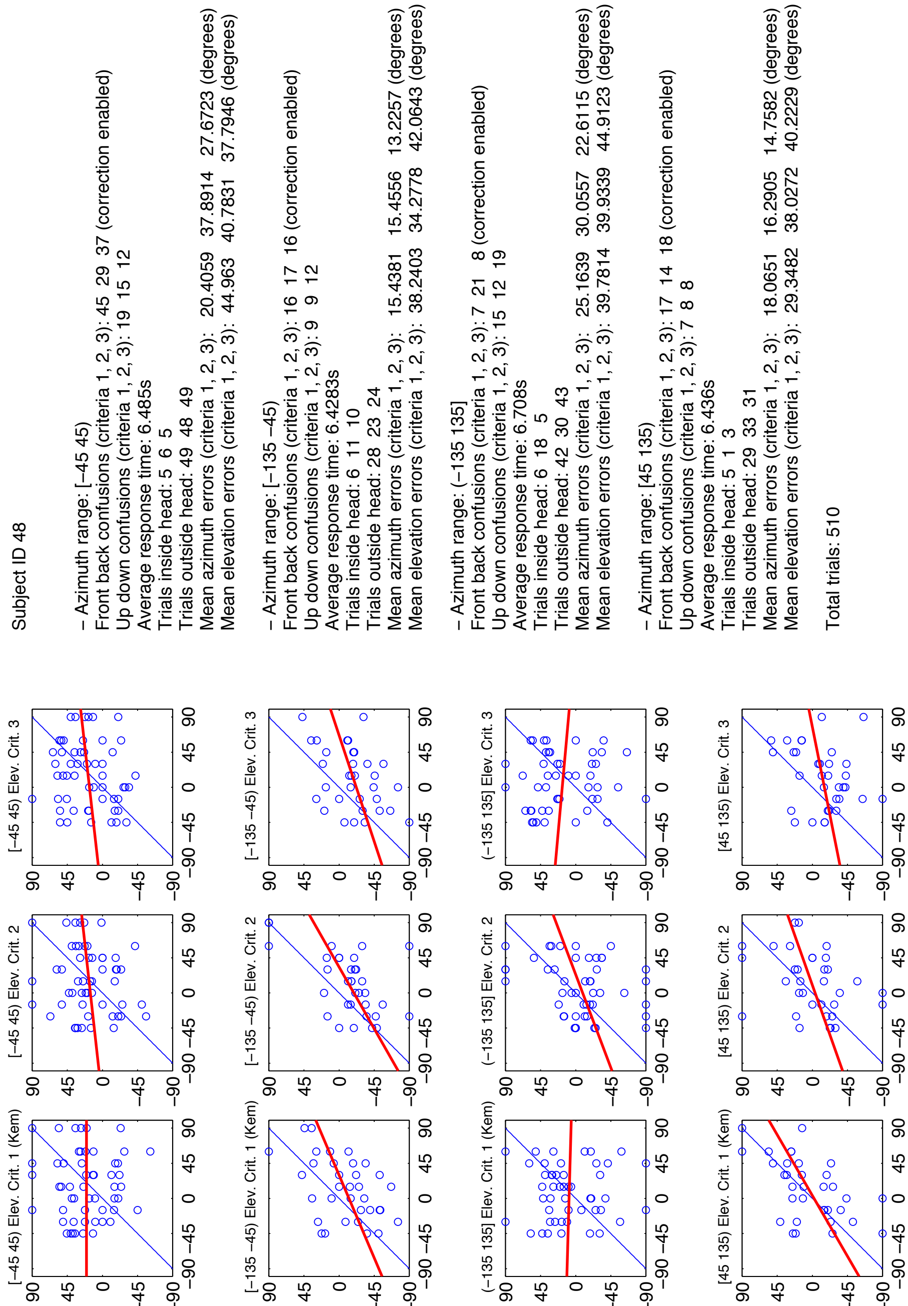
Results split by sectors confirm the good performances of the second criteria for this subject. However it is interesting to note how the KEMAR is significantly better (33%) than the third criteria in the back area. Another interesting point regards how there is a complementary situation between the left and the right section: while the azimuth is better on the left, the elevation is better on the right. This could confirm that there is no correlation at all between azimuth and elevation cues.

Subject 48

This subject decreases elevation performances of 5.4% in respect to KEMAR.



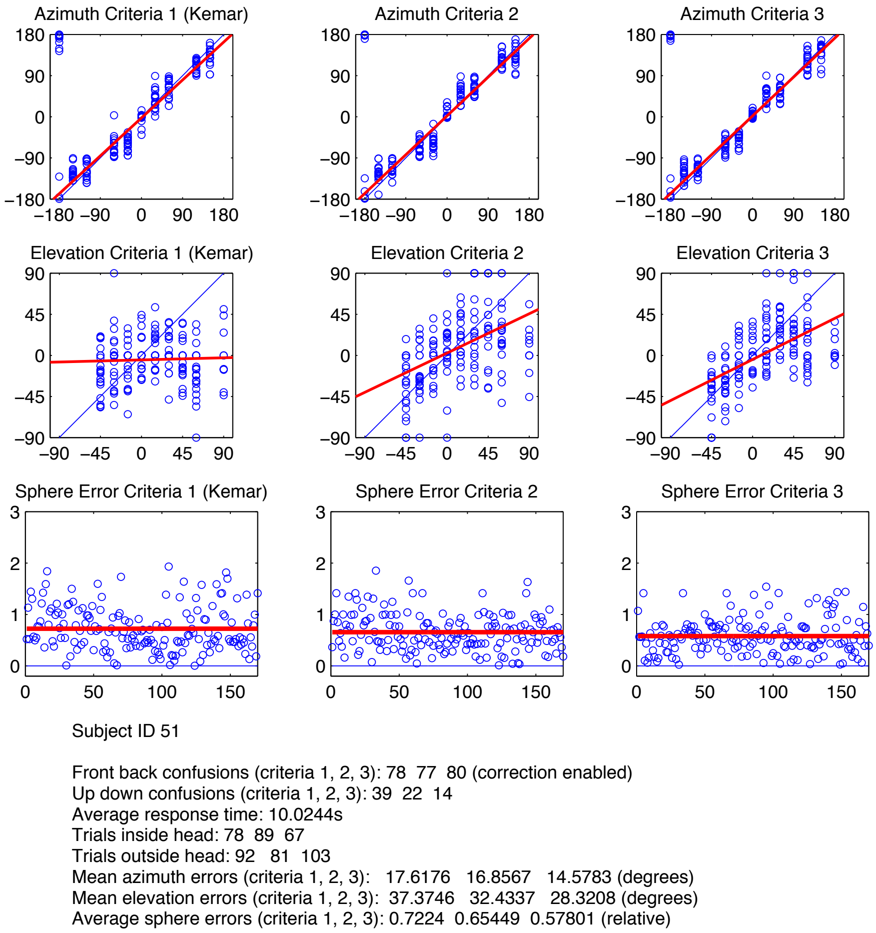
This first plot does not indicate a particular HRTF as the best one in elevation tasks for this subject. As previously stated for subject 44, this is also a naive subject and maybe he or she has a slow-growing learning curve for this type of tasks. Response time is in average and externalization is good. However, the second plot could help to understand why the third criteria is globally a bit worse than the other two.



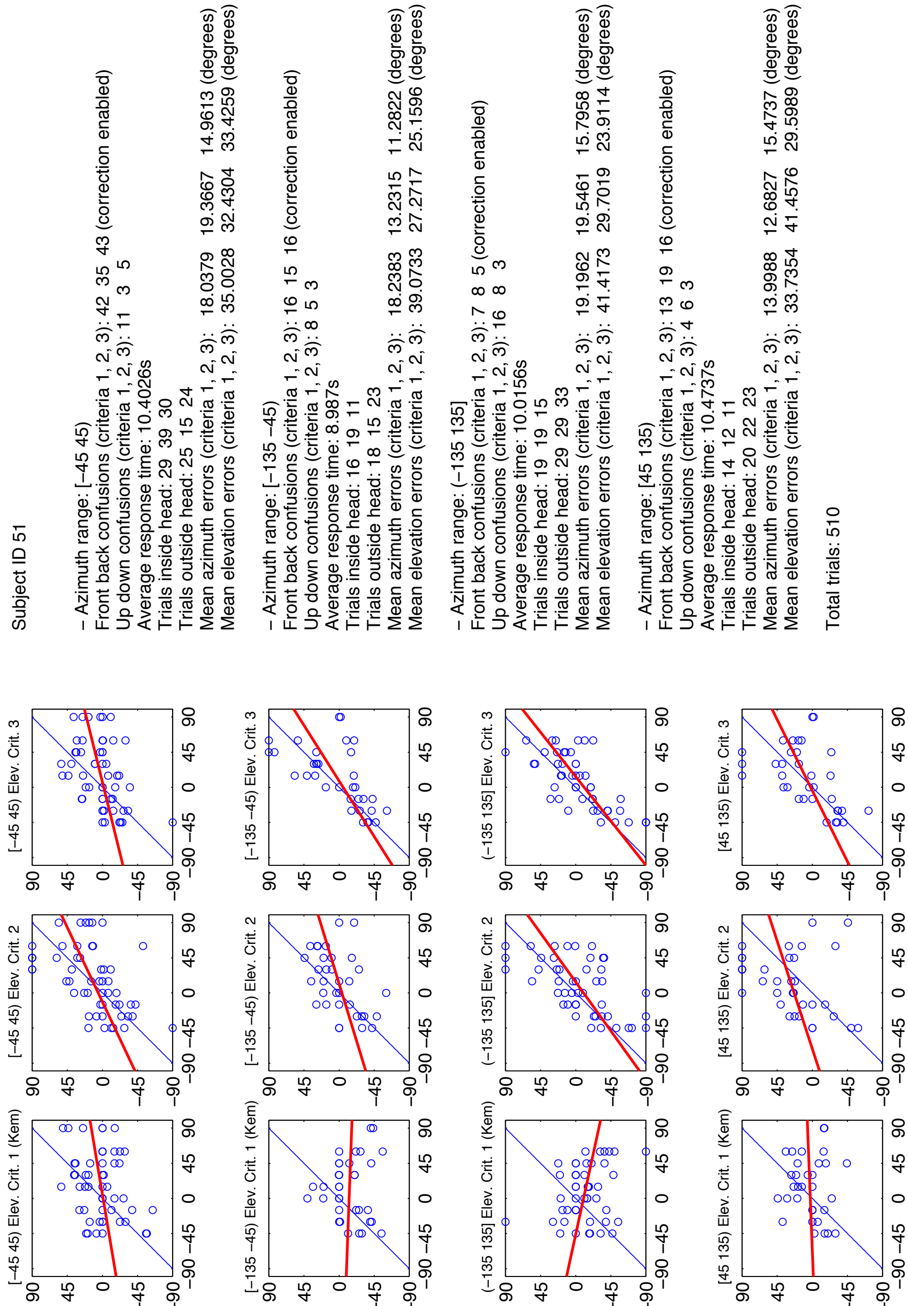
It is easy to be captured by the strange trend of the third plot of the third row (criteria (1, 0, 0), rear area). This plot has an error of 45 degrees in elevation that means completely random answers. If we consider only the second half of the experiment (the plot is not shown here) the results have still large absolute errors but they are relatively smaller. However also the second half of the experiment confirms a completely outlier trend for the back region of the third criteria meaning that the selected HRTF is totally inappropriate for that subject but only on the back area. The chosen position of the focus point during the tracing operation or the particular most external contour traced on the pinna could have caused this result.

Subject 52

This subject increases elevation performances of 24% in respect to KEMAR.



This subject was not planned to be included in this work and consequently his or her results are not included in the summarized results. This is because the experiment was performed at a late stage of the work. However the very interesting outcome drove us to add him or her here. In fact there is a significant improvement on elevation between the KEMAR and criteria (1, 0, 0). The KEMAR linear regression is an horizontal line, while the linear regression of (1, 0, 0) is dramatically better. The third criteria also decreases up/down confusions and increases externalization. Azimuth results are also really coherent and precise.



The same highlighted trend is also visible splitting azimuth on different ranges. The poor performances of the KEMAR are particularly visible in the rear area, where the linear regression shows a completely reverse trend with almost random results. Even if the KEMAR gives the worst performance in the rear area, criteria (1, 0, 0) gives the best performance in the same area.

5.2 multimodal experiments results

General preliminary considerations

We would like to present these test results as a starting points for future works. These works may consider the Testbed evaluation method. This is based on taxonomy, outside factors (task, users, environment and system) and performance metrics that gives quantitative performance results and heuristics/guidelines to produce user centered applications [9]. However costs and benefit of this evaluation approach depend on the complexity and maturity of the application.

As another preliminary consideration, previous haptic exploration tests showed as users performances have a better success rate when they exploit a more regular exploration strategy than the other users [11].

Another important point is that the orientation of the haptic device, with respect to the tablet orientation and to the head orientation, is fundamental. While PC users can quickly and intuitively fix uncorrected orientation of the mouse, this is not possible for blindfolded or blind subjects. For this reason we told the subject to maintain the head in a fixed frontal position and placed tablet and mouse on a comfortable and straight position just on the right (or on the left depending on right or left handed preference) of the subject's head. Previous works cited above show as an object can be perceived as rotated when the device is not aligned to the tablet.

Our results show micromotion (small precise movements) and macromotion (macroscopic quick changes on the map) TAMO movements. Micromotion in usually performed in order to maintain a level of stimulation to the receptors and macromotion in order to seek features of interest and orient objects. This difference has been originally discovered on eyes movements [63] but it is also valid in haptic exploration.

5.2.1 Experiment 0

ID	Time (only TAMO)	Time (only Audio)	Time (TAMO + Audio)	Improv. TAMO/TAMO+Audio	Improv. Audio/TAMO+Audio
29	10.8	12.6	8.1	25%	36%
30	8.7	10.6	6.5	25%	39%
36	8.4	8.4	7.5	11%	11%
43	11.7	10.4	6.2	47%	40%
44	13.4	11	7.9	41%	28%
50	10.4	20	8.4	19%	58%
51	9.1	8.2	7.2	21%	12%
52	11.2	10.6	8.2	27%	23%
53	13.5	11.4	8.3	39%	27%
54	13.8	13.7	11.9	14%	13%
55	12.7	9.7	10.8	15%	-11%
56	18.9	8.8	7.8	59%	11%

Table 6: Multimodal Exp 0 Summary Results. The combination of TAMO and Audio dramatically increases the performances.

Results analysis of exp 0

Table 6 summarizes the results of the first experiment *(expo)*. Their are also shown on figure 28. We decided to compare the absolute time needed to reach the marker goal, at the center of the virtual map as a first metric. We found an average improvement between TAMO

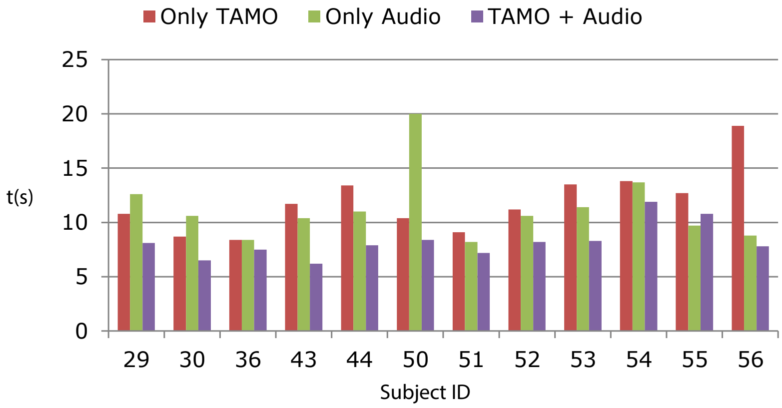


Figure 28: Multimodal Exp 0 times. This figure plots the results shown on table 6.

and Audio + TAMO of 28% with a peak of 59% and a minimum improvement of 11%. Standard deviation calculated on improvements is of 15%. There is also an average improvement with Audio + TAMO of 27% in respect to only Audio. As shown on the figure, only Audio usually performs better than only TAMO but this could be related to the particular elevation increment used on the TAMO device i.e. the elevation decreases with the square of the distance from the center. Additionally there could be a different law capable of a better control of the haptic feedback.

These results are significant confirming a strong and effective multimodal integration between TAMO (haptic device) and spatial audio. This could be related to the audio capability on providing a constant global information about the position on the map, while TAMO provides just a local information. Combining the two feedback, the subjects can easily reach the goal. This possible explanation is confirmed by those trials with only haptic feedback: the subject can hardly reach the central area of the map and when it is reached, he or she can easily complete the experiment. On the other hand, on trials with only audio feedback, the subject can easily reach the central area of the map but he or she needs some time in order to properly place himself or herself above the marker. Appendix A contains all the trials performed on all the subjects, with each exploration paths. Additional and more rigorous analysis will be performed on this data, however it is clear that the exploration paths with both TAMO and 2D Audio are less tangled, at least by a first qualitative inspection.

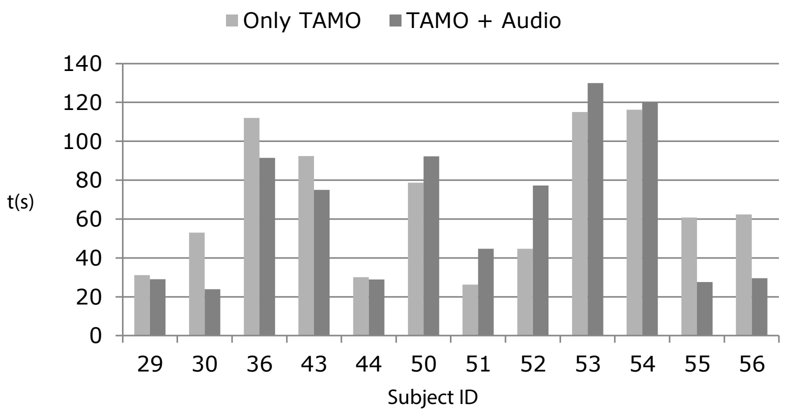


Figure 29: Multimodal Exp 1 times. This figure plots the results shown on table 7.

5.2.2 Experiment 1

ID	Time (only TAMO)	Time (TAMO + Audio)	Improvement/Decrement
29	31.2	29	7%
30	53	23.9	55%
36	112	91.5	18%
43	92.4	75	19%
44	30.1	28.9	4%
50	78.7	92.2	-17%
51	26.3	44.7	-70%
52	44.7	77.3	-73%
53	115	130	-13%
54	116.2	119.9	-3%
55	60.8	27.6	55%
56	62.4	29.6	53%

Table 7: Multimodal Exp 1 Summary Results.

Results analysis of exp 1

Table 7 summarizes the results of the second experiment *(exp1)*. Their are also shown on figure 29. The average reconstruction time is of 68.6s without audio and of 64.1s with audio, with a difference of 6.6%, that is probably not so relevant since the standard deviation of data is really high (36s). Note that failed reconstructions gets the maximum execution time of 150s although failed reconstructions are really sporadic: for a complete list of results see appendix A. As a general consideration, the prism with triangular base is the easiest figure to be reconstructed while the cylinder is the most difficult one. The most effective exploration approach consists on exploring the map moving on a grid with vertical and horizontal lines. A large number of subjects preferred instead to follow the contour of the figure but it is not easy to understand if the current movement is on straight direction or on a diagonal one. These results probably suggests that there is not a strong role for audio in reconstruction tasks. However it is still interesting to analyze the exploration paths followed by the subjects and try to understand if they changed with or without audio feedback. Additionally it is interesting to correlated changes in the exploration strategies according to the instants in which subjects give wrong answers. For the last 4 subjects we collected these data (subjects ID 53, 54, 55, 56) and they are shown on table 8. It is interesting that the subjects proposed the rectangle quite often as a possible answer. This could be related to a different perception of x and y dimensions. In addition, subjects always refer to figures as 2D shapes instead of 3D shapes: TAMO seems not able to provide an elevation feedback as large as an horizontal movement.

ID	Square no audio	Triangle no audio	Circle no audio	Square audio	Triangle audio	Circle audio
53	1.30 rhombus	-	1.21 rectangle	-	1.16 circle	1.55 square
	1.40 ellipse		1.34 square			
			2.18 triangle			
54	0.32 circle	-	1.18 square	-	1.58 rectangle	1.10 triangle
						1.55 trapezoid
55	-	0.45 circle	-	-	-	-
		1.42 square				
		1.53 rectangle				
		2.03 exagon				
56	0.58 rectangle	-	0.34 rectangle	-	-	0.42 square

Table 8: Multimodal Exp 1 Wrong Answers with times in *mm.ss*.

5.2.3 Experiment 2

ID	Error px (TAMO)	Error px (Audio 2D)	Error px (Audio 3D)	Num size err (TAMO)	Num size err (Audio 2D)	Num size err (Audio 3D)
29	60.25	62.5	75	4	2	1
30	33	54.25	29.5	3	3	1
36	57.75	50.75	59.25	1	2	2
43	-	-	-	-	-	-
44	98.25	68.25	43.5	2	4	2
50	86.5	80.25	85	4	2	4
51	62	61.25	27.5	4	3	1
52	66	25.75	25.5	3	3	0
53	125.25	82.75	45.5	3	3	2
54	56.5	61.25	71.75	2	2	2
55	88.75	127.75	169.25	4	2	3
56	65.75	47.75	48	2	2	3
tot	800	722.5	679.75	32	28	21
mean	72.7	65.7	61.8	2.9	2.5	1.9

Table 9: Multimodal Exp 2 Summary Results. Audio 3D helps both on localization and sizing tasks.

Results analysis of exp 2

Table 7 summarizes the results of the third experiment *(exp2)*. These data are also shown on figure 30 (displacement error) and on figure 31 (size errors). The improvement on displacement task between only TAMO and TAMO + 3D Audio is of 15.0%. Improvement on displacement error between TAMO and TAMO + 2D Audio is of 5.9%. Regarding the number of size errors with 3D Audio it decreases of 35% in respect to only TAMO. It is notable to see how 3D audio dramatically decreases the number of size errors since the subject was not aware of the presence of 2D or 3D audio and all the subjects affirmed that they did not consciously note the difference between the 2 audio

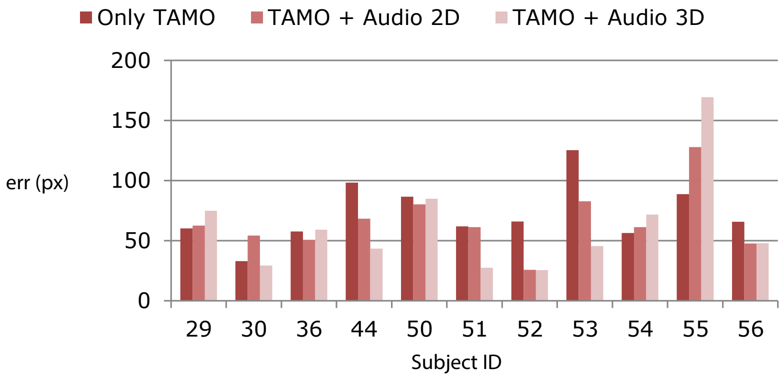


Figure 30: Multimodal Exp 2 displacement error. This figure plots the results shown on table 9 regarding the displacement of the cubes.

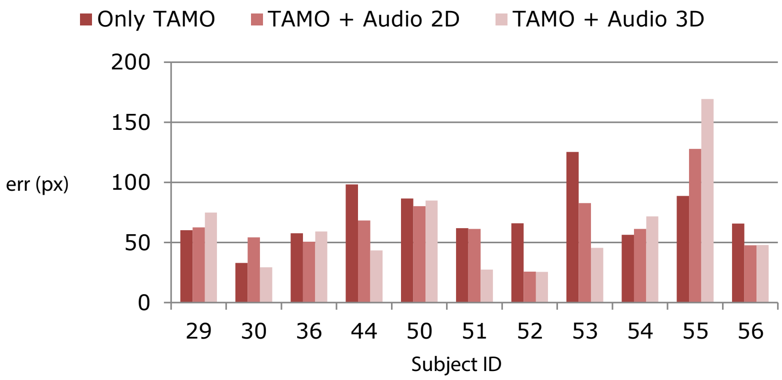


Figure 31: Multimodal Exp 2 size error. This figure plots the results shown on table 9 regarding the sizes of the cubes.

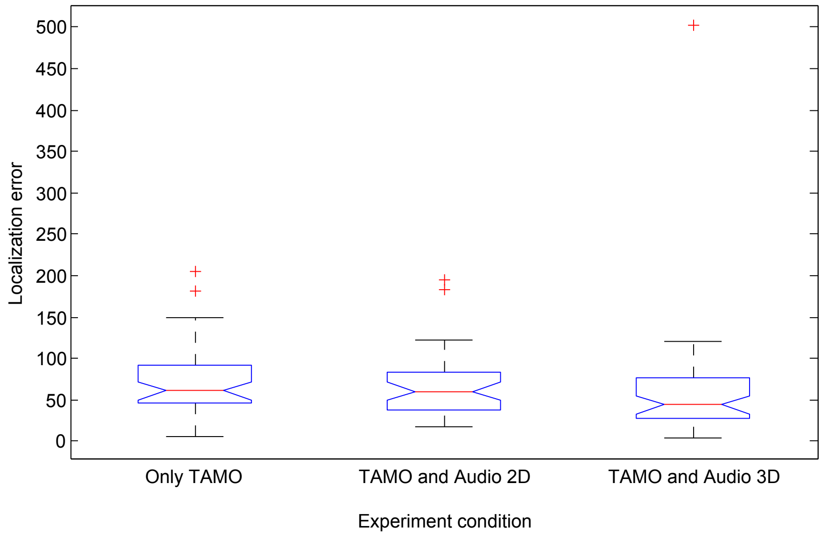


Figure 32: Multimodal Exp 2 Kruskal-Wallis nonparametric One-way ANOVA. The plot shows the localization error with the 3 criteria plotted using the Kruskal Wallis test. Percentiles (25 and 75) are also shown. Significance level is $p = 0.0664$.

modality. This is also proved seeing as they always refers to the figures as 2D figures ("squares") instead of 3D figures ("cubes"). It should also be noted how the subjects usually tend to *underestimate* the size of the squares/cubes: the smallest proposed size (45px) was chosen 30 times, while the largest size (185px) was chosen only 9 times on a total of 132 trials, recalling that both sizes were not really proposed: the only two sizes proposed were 80px and 150px as explained in 3.3. This could be related to the relative TAMO size compared to the tablet size: a minimum physical movement of the TAMO correspond to a large virtual movement on the map

Audio (2D and 3D) also reduces the time of the first contact with the square/cube: from an average of 8.7s (no audio) to 7.7s (audio) with a decrement of 11.5%.

As a final analysis on figure 32 we show a Kruskal Wallis non-parametric one-way ANOVA[1]. The test does not reach a complete significance level ($p = 0.0664$) however it shows a trend where the displacement error decreases from the Only TAMO condition to the TAMO + Audio 2D and to the TAMO + Audio 3D condition.

1 This plot was kindly provided by Luca Brayda and Claudio Campus from IIT.

6 CONCLUSIONS AND FUTURE DEVELOPMENTS

Psychoacoustic Conclusions

In the first part of this work, the use of pinna reflection pattern analysis as an HRTF selection method looked promising and our experiment confirmed these expectations. Compared to the use of a generic HRTF with average antropometric data, the pinna reflection approach increases the average performances of 10% with a peak of 34%. The system also significantly enhances the externalization and reduces the up/down confusions.We also showed as the most external contour (C1 or C2) is the most important and robust in elevation localization tasks with this setup. The entire experiment was conducted on mean conditions (e.g. with *non-individual* headphones compensation) in order to create a possible future scenario for practical applications. We did not only show the results of this psychoacoustic experiment but we also discussed a reliable and tested environment to conduct similar experiments, providing the code and the main algorithms to perform them. The system is also capable to automatically plot the data in order to quickly revise them after the experiment. It runs just on a single PC with Matlab, Pure Data and headphones. Since the significant amount of benefit introduced by such a low-cost tracing method, future works should consider the role of pinna reflection patterns in the HRTF selection for elevation localization tasks.

Multimodal Conclusions

On multimodal experiments we discovered different integration levels according to the specific required task.

In the first experiment (reaching task) we observed a perfect multimodal integration since there is a combination of a global feedback (audio) and of a local feedback (haptic). All subjects significantly increase their performance with both haptic and spatial audio with an average of 27.5% proving that this combination is really effective for these kind of tasks. Since the required devices for this application are low-cost and since it is easy to create virtual maps (also extracting them from existing *CAD* planimetries), real application as helpers for blind people should be considered.

The second experiment (object reconstruction) does not seem instead to receive a particular benefit from the presence of audio and additionally the TAMO device does not seem adapt for these tasks. Further investigation must be performed on the available data, different devices should be considered and additional experiments should be conducted on larger tablet.

The third experiment (object displacement and size estimation) receives a significant benefit from the presence audio feedback: spatial audio reduces the first contact time, increase the localization performances and 3D audio also decreases the number of size errors. This system is therefore a good starting point to simplify the orientation in complex virtual maps for blind people.

Future developments

We would conclude this work providing some future developments.

One of the most relevant aspect of this work is the pinna extraction. We saw that the position of the focus point significantly influences the computation of the f_0 values. Our current solution locates the focus point as a single position and it does not take into account a rectangular area with the idea to repeat the calculation of f_0 varying the focus point position inside the rectangular region. A really similar minimization problem has also be presented in [57]. For each position we could calculate the mismatch with CIPIC subjects and then we could select the subject that minimizes all the mismatches calculated with the different focus point positions. Alternatively we can use a reverse approach and extract contours form the CIPIC database, starting from the f_0 values. Then we can try to match the manually traced pinna contours with those extracted from the HRTF. We can also try to use the three different contours in a tuning process: one contour could be used to select a subset of HRTF and the remaining contours could be used to select smaller subsets starting from the first one. Contours have different weights and they could play different roles in the selection. It should be also interesting to develop a tool that automatically traces the contours from a pinna image. The entire tracing systems currently works on a 2D image: a 3D representation of the ear would open to new opportunities since the pinna does not lean on a plane parallel to the median one. This is particular important for subjects with protruding ears. Different input method for subjects in the psychoacoustic experiment could also be considered e.g. a sort of joystick or helm in order to insert azimuth and elevation values. Finally the role of training session that we found in this psychoacoustic experiment should be further explored and understood. One different but still interesting research area could be the adaptive HRTF selection and the adaptive map exploration modality addressed using machine learning: rather than using a priori rules to select HRTF we can potentially exploit user's results watching his or her behaviors in the sensed context [29] [48].

On multimodal experiments a pletora of possible developing paths could be explored: different maps with a different number of figures, different number of sound markers and obstacles to be avoided. In general we can think about including high-level cognitive, problem-solving tasks that are specific to the application domain [9] such as the creation of a path from a starting point to an exit point. It could be interesting to ask the subject to walk blindfolded in a real room after an exploration session. There is also the possibility to constrain the haptic exploration only on certain directions or areas [37], configuring the experiment as a navigation task. It should be also interesting to see performance comparison in multimodal environments between blind people and blindfolded. Additionally, the multimodal environment could be tested with the TAMO device and a larger tablet area or with a different haptic device such as the already mentioned Phantom. However, before moving on these research aspects, the already collected data must be further analyzed in order to better identify possible future directions.

7 ACKNOWLEDGMENTS

This Master Thesis is not only the result of knowledge acquired during my career as a student, but it is also the result of a personal growth for which I would really like to thank all the people who helped me during this travel.

In particular I would like to thank the research group of Sound and Music Computing at the University of Padua for its help and sustain during these months: Professor Federico Avanzini for his advices, he was a solid and concrete point of reference for me; Michele Geronazzo for his key guidelines: in this work he was as a fundamental mentor for me; Simone Spagnols for the sustain and for all the hints and responses he gave to me, he was a really helpful aid. They kindly sustained the huge amount of emails and interruptions I provided them. At the University of Padua, they are an amazing research group with a strong historical tradition on sound and music computing, started by Professor Giovanni De Poli. I am honored to have been at least a little piece of this story.

Special thanks also to Luca Brayda and Claudio Campus from IIT for their ideas and their feedback on multimodal experiments.

I am also in debt with all the subjects who participated to the experiments described in this work. They found the time for me and for this work in their agenda, often with a great enthusiasm, and they did no ask for anything in return. I will not forget this.

I would also like to thank Dr. Marco Bressan, my advisor of the BS thesis. I did not include acknowledgments on my BS thesis, so this is the right moment to thank him too. He did not only helped me on my thesis but he also gave me an incredible amount of really helpful practical advises. He was also able to motivate me during the entire work.

There are other teachers who deeply influenced my career as a student and I would be a different person without them.
The first one is Maestro Beppe Nordio. He was my elementary teacher of maths. As time went by, he also became a friend and he gave me the possibility to help him as chess instructor and web developer instructor. He passed away some months ago but there are a lot of circumstances that remember me about him so I always feel him as if he was here.

The second one is my High School Professor Rosa Maria Zermian. She was a great (but also demanding!) teacher. She told me to do not give her this thesis written in English because "I could not understand anything!" - she said. Now I'm writing these words in English and she will need to toil a bit in order to translate this: I'm taking a little revenge for all the latin texts she asked us to translate!

Finally I would like to talk about my family and my friends, and I think it is time to leave the English language and to write on my mother tongue that is the language that better allows me to convey my feelings for them.

FAMIGLIA E AMICI

Una parola speciale per la mia famiglia: a mamma e papà che hanno sempre creduto in me e hanno sempre dato tutto perché io avessi il meglio. Grazie, grazie, grazie.

Tutti i parenti, che si sono sempre interessati a me con grande affetto, felici per i miei risultati negli studi. I nonni, che sempre mi hanno voluto tanto bene e ai quali voglio tanto bene anch'io; quelli che ci sono e anche il nonno Orfeo che non c'è più, ma che tanto avrebbe voluto esserci. La Zia Ottorina, che mi ha sempre amato come fossi un figlio.

E poi gli amici.

Quelli della "***Tavolta RotOnta***".

Grazie a ***Pallina***, che sei uno strano personaggio rosa e rotolante, la cui strada sia è intrecciata con la mia ben vent'anni fa e da allora, tra alti e bassi, ci sopportiamo, aiutiamo, incoraggiamo e fraintendiamo. Senza di te sarei cresciuto in tutt'altro modo e probabilmente questo fantastico gruppo di amici con cui esco a Vicenza non esisterebbe così com'è. Le tue idee pazze e assurde, qualche volta sensate, qualche volta proprio no, il mio cercare di fermarti e farti ragionare, il tuo cercare di spingermi a non farlo, le indimenticabili feste in casetta, le persone che abbiamo visto arrivare, le persone che abbiamo visto andar via, le fotografie, le cose confidate, i computer che non funzionano, le corse assieme, i nostri "passo a prenderti alle ore x":, sono cose che non voglio mai dimenticare.

Grazie a ***Bax***, che se quella volta non ci fossimo incontrati in piscina, chissà come sarebbe andata, ma ormai è acqua passata e, in fondo, va bene così.

Grazie a ***Vero***, che ogni tanto mi domando "come cavolo faremmo senza di lei?!". Da quando ti ho incontrata ci sei sempre stata, puntuale, precisa, efficiente e soprattutto sorridente.

Grazie a ***Chiara***, e un po' anche le tue mille preoccupazioni per lo studio, ma più che altro il tuo volermi bene. Se ora ho un orso gigante in salotto è merito tuo.

Grazie a ***Silvia*** e il nostro contenderci quel tipo rotolante: in fondo, lo so bene, sei un po' gelosa. (ma shhh, è tutto tuo, davvero!).

Grazie a ***Greg*** e la tua smisurata cattiveria sempre accompagnata da sguardi inquietanti e da un sorrisone. Tu, le tue birre, la tua porchetta, il tuo essere altruista senza nulla chiede in cambio. Sei una persona fantastica con la quale passerei le ore a chiacchierare. E, per inciso, sei stato anche un ottimo soggetto sperimentale.

Grazie a ***Cox*** con il tuo essere festaiolo, per aver trasformato una vacanza al mare in una settimana esilarante mischiata in un cocktail a base di rientri all'alba, Texas e caviglie slogate. Indimenticabile.

Grazie a ***Nic*** perché sei Nic e mi fai sempre sorridere. Con i tuoi messaggi improbabili ad ogni ora "facciamo un poker?", i tuoi abbordaggi alle ragazze, la tua compagnia e le tue pazze idee.

Grazie a ***Co*** che rimarrai sempre per me la mela proibita, nonostante gli inviti a cena, i gabbiani, i tramonti ecc. ecc. Una delle amiche storiche, con cui ho condiviso tante sveglie all'alba per prendere la corriera e che ora ogni tanto vedo puntualmente sparire in sessione d'esami. L'importante è che tu sappia sempre che ti voglio bene e che a pallavolo sono chiaramente più forte di te.

Grazie ad ***Alessandra*** e i tuoi sorrisi ma soprattutto il tuo non voler correre rischi "che poi gli altri se la prendono". Però va benone così, che chissà quante figuracce ci ha evitato (e ci eviterai!).

Grazie a ***Rasso***, in fondo gli scacchi possono trasformare incontri casuali in amicizie che poi dureranno, chissà, una vita. Una lunga vita in cui ti chiederò indietro, un pezzo alla volta, tutti i panini.

Grazie a ***Ste*** uno storico amico della compagnia e penso di poter dire di averci visti crescere e cambiare (in meglio si spera!) assieme in questi anni. Ne abbiamo passate così tante che ripensandoci la prima cosa che mi passa per la testa è: ne è valsa assolutamente la pena.

Grazie all'***Onto*** che se anche ora non esci più insieme a noi, l'hai vista nascere questa compagnia di amici. E io serbo sempre un bellissimo ricordo dei momenti passati con te: i kebab, i calcetti, le pagelle, le tue risate e le partite dei mondali.

Grazie a ***Ila*** che quando ci sei mi (e ci) accompagni sempre con la tua assoluta cattiveria. Ma dopo tutto è giusto così: altrimenti non saresti tu. E ricorda che la telecamera va tenuta ferma quando si inquadra, cribbio!

Grazie a ***Cipari***, sei na bronsa coerta, senza alcun dubbio. Mi hai invitato così tante volte in bruschetteria e alla fine sono venuto solo una volta: come buon proposito devo rimediare!

Grazie a ***Pellah*** e le discussioni con te di Matlab, C, Linux e quello che capita. Ma poi parliamo anche di cinema e qualche volta di viaggi! È esilarante confrontarsi con un fisico su queste cose!

Grazie a ***Fify***, nel bene e nel male ne abbiamo passate tantissime assieme. Oggi ti laurei insieme a me quindi non posso che farti un grandissimo in bocca al lupo per il tuo futuro.

Grazie a ***Saretta :-)*** un'amica fantastica, sai sempre come rendermi felice e farmi sorridere. Anche quando non ci vediamo per un po', non perdiamo mai i contatti e sai essere molto affettuosa. Due tuoi messaggi sono tra i più belli che abbia mai ricevuto: li conservo con grande affetto. Ricorderò sempre la sera in cui ci siamo incontrati per la prima volta, alla festa della partenza di Cipe, e il nostro metterci a giocare con i tappi delle bottiglie. E poi, dopo tanto tempo, eccoci qua! Ti voglio bene.

Quelli di "***All DEI Long***".

Grazie a ***Vanuz*** il mio PR preferito, sempre pronto ad organizzare. Mi diverto sempre ad intrattenere conversazioni con te di checavoloneso, roba Apple, orologi, moda o quello che capita!

Grazie a ***Emma***, ti ho scoperta un po' alla volta in questi anni e ho capito come stare bene con te conoscendo man mano i tuoi pregi e i tuoi difetti, come tu hai fatto con i miei. Mi hai cambiato non poco, anche musicalmente parlando, pitonica ragazza!

Grazie ad ***Ale***, il grande gigante, abbiamo condiviso assieme la laurea triennale e da allora ne sono successe di cose. Sei scappato in Danimarca ma per questa giornata di festa mia, di Colla e di Giovannino sei tornato, quindi non posso che ringraziarti e spero di non pentirmi di queste parole dopo aver ricevuto da te e dalla tua manina tante botte ai festeggiamenti.

Grazie a ***Marty***, il mio Amico, con la o minuscola e la A maiuscola, anche se sei una ragazza. Con te gli scambi di consigli sono una costante e il tempo vola davvero quando siamo in compagnia. La notte prima di partire mi hai raccontato che potresti pensare di restare per sempre in Danimarca dopo la tesi. Io ti auguro di realizzare i tuoi sogni ma se questi ti porteranno all'estero, lontano da casa, sappi che mi mancherai e non mi accontenterò di vederti ogni tanto su Skype: serviranno dei ritrovi! Ah, già: comprati un kit di pulizia lenti, diamine a te.

Grazie a ***Nico***, stai pur certa che non dimenticherò mai il quaderno che mi hai regalato, con la raccolta di quattro fantastici anni di università insieme. Sei stata un bravo capitano e ti stimo tantissimo per il tuo coraggio nel voler inseguire i tuoi sogni. Lo sto cercando anch'io e spero di farcela. Non mollare mai.

Grazie a ***Giovannino***, sei un pazzo totale e nonostante gli avvenimenti passati non l'ho dimenticato. Un giorno lo ricorderemo brindando assieme, davanti ad una Guinness, una di quelle vere, stanne certo.

Grazie a ***Colla***, ci siamo conosciuti alla fine dell'Università e questo è stato un vero peccato: chissà quante altre esilaranti storie avremmo avuto altrimenti da raccontare. Ma non ti preoccupare: non ci perderemo di vista. E sappi che continuerò a romperti con messaggio del tipo "5x03" seguiti da un commento breve e incisivo.

Grazie a ***Feffe*** non dimenticherò mai le prime lezioni di analisi 2 insieme a te. Tu sei la persona che ha dato a tutti noi quel più totale e assoluto senso di "noncelafaremomai". E invece eccoci qua! Penso che mi ricorderò per sempre quella mattina dopo il Botellon, quando mi sono svegliato a casa tua, solo che tu non c'eri e non c'era nessun'altra persona. Robe da matti. Ho recuperato le mie cose e mi sono diretto verso casa della Marty, passando per un Prato della Valle che era così assurdamente silenzioso, dopo tanto rumore.

Grazie a ***Baruz*** e tutte le volte che ci hai ospitati da te per una pizza, una serata in compagnia o per complottare qualcosa. Per i tuoi gusti culinari, la cena di pesce con autista che ancora mi devi, la tua maniacale sicurezza, le tue battute sempre pronte, il tuo essere Project Manager (qualche volta un po' troppo pesante a dire il vero!), i tuoi spostamenti i bici, il nostro viaggio in California, le nostre giornate al DEI. E chissà quante altre cose ancora in futuro.

E poi gli altri amici...

Grazie ad ***Ale Urso***, il grande gigante gentile, sei sempre disponibile e innamorato di qualsiasi cosa abbia una mela morsicata. Come dimenticare le nostre chiacchierate in palestra?!

Grazie ad ***Alice***, che spesso ti smazzi per organizzare e farci ritrovare, anche se ultimamente non sono stato molto presente, lo apprezzo tanto lo stesso perché so bene cosa vuol dire farlo. Spero nel post laurea di essere più libero.

Grazie ad ***Elena*** che se anche guardo dentro all'obiettivo ti voglio bene lo stesso sai?! Con me sei sempre molto ospitale, soprattutto quando ti presenti in pantaloncini grigi. A proposito: ho ancora un video da montare, non me ne sono dimenticato!

Grazie a ***Betta***, ho appena scoperto le tue origini e con te non parlerò più fino a quando non mi sistemerai quel dosso, dannazione!

Grazie a ***Rosi***, voglio un po' di bene anche a te, nonostante tu sia un maledetto e incallito Android user.

Grazie a ***Colo***, per avermi fatto credere di essere un vero nobile e tuttora non so ancora se sia verità o invenzione. E poi grazie anche

per le tue rime, sebbene ultimamente tu stia un po' perdendo la stoffa, o sbaglio?!

Grazie a ***Giulia*** per avermi affiancato nella prima avventura cinematografica! Mi sa che dovrò mandarti copia della tesi online in Spagna: fai tanta festa!

Grazie a ***Martina AmarTiMara*** hai reso un sogno realtà. Il tuo modo di scrivere e le tue storie sono bellissime. E mi trovo davvero bene a lavorare con te. Spero che il nostro primo lavoro assieme sia piaciuto anche a te e di non aver deluso le tue aspettative. Non puoi neanche immaginare quanta voglia abbia di rimettermi dietro ad una macchina da presa per raccontare a tutti una tua storia.

Grazie a ***Gian***, per avermi sopportato come compagno di stanza in Galles e per le nostre conversazioni mattutine "Mmm?" "mmm!" "mmm.". Sei anche un ottimo grafico e hai una bellissima squadra di service. Mi piace fotografare i vostri eventi. Continua così e in bocca al lupo anche per la tua tesi!

Grazie ad ***Ombre***, potrei chiamarti la guida automobilistica umana. Sempre gentilissimo e disponibilissimo quando si tratta di dare consigli su come raggiungere una meta o evitare il traffico verso la spiaggia. E poi lo sai che apprezzo la tua cattiveria, quando si tratta di scrivere papiri o fare scherzi. Spero solo che quando leggerai queste parole non mi avrai fatto troppo male!

Grazie a ***Cipe*** sei il mio antieroe per antonomasia, la mia nemesi. Potremmo insultarci per ore solo per dilettare il pubblico o per il gusto di farlo. A questo punto potrei insultarti in tanti modi, sceglierò il più cattivo: chi hai baciato nell'estate 2012?!

Grazie a ***Scalchi*** sei cambiato tantissimo dai tempi del Liceo ed è bello essere rimasti sempre in contatto da allora ed esserci visti a feste e concerti. Sia chiaro, non sto parlando di cambiamenti fisici: per quello rimani sempre il solito brutto Scalchi. Ma giusto così!

Grazie a ***Virgi*** ti ho conosciuta e incontrata per caso su un set fotografico e poi ho scoperto non solo una modella ma anche e soprattutto un'amica. Un'amica che deve imparare a scrivere sms, perché quando lo fa metà delle volte non li capisco e l'altra metà poi non si ricorda quello che ha scritto! Però noi ci scherziamo su, e allora alla grande così!

Grazie a ***Il Bosco*** perché non so mai stato così felice di aver regalato un componente elettronico a qualcuno e un po' mi piace pensare che quando sarai a Mountain View lo terrai sempre sulla tua scrivania e ricorderai quell'imbecille vestito da Babbo Natale che te l'ha regalato.

Grazie a ***Nicola Baraldo*** un po' il pessimista del corso di laurea magistrale. Ricorderò sempre i messaggi che mi mandavi durante gli esercizi preliminari per l'Erasmus in Galles! E ricorderò anche sempre le nostre discussioni sul canto e i musical!

Grazie a ***Cecca*** ma anzi più che ringraziarti un po' ti odio, perché invece di presentarmi la tua amica, ti ci sei messo assieme! (nice move). Ma sono sicuro che troverai il modo per redimerti: mandami un frammento di codice del kernel di Mac Os X, quando lavorerai in Apple!

Grazie a ***Elena Zanotto*** ti cito con nome e cognome per non confonderti con l'altra Elena e anche perché so che a te piacciono le cose fatte in modo preciso! Ti sei così tante volte offerta per fare gli esperimenti della tesi e mi hai chiesto così tante volte di bere un caffè durante una pausa studio ma poi alla fine non ci siamo quasi mai riusciti. Ce la faremo. E una sera riuscirò a vederti uscire anche a Vicenza, non solo a Padova, perché ora come ora non riesco proprio ad associarti alla città delle "e" aperte!

Grazie a ***Luci*** sei stata una persona importante per me in questo ultimo anno e spero di essere stato altrettanto: *"non troppo cozzo e non troppo distante"*, come hai scritto una volta. Sei una ragazza stupenda e sensibile, iper attiva e dalle mille risorse: con queste tue qualità e con l'aiuto di chi ti vuole bene, ti assicuro che saprai superare tutte le difficoltà. Non smettere mai di crederci.

Grazie a ***Giulio***, il Telecomunicazionista, quando mi chiedo se c'è qualcosa da aggiungere oltre a quanto già scritto nella dedica di questa tesi, mi rendo conto che quella dice già tutto: *Sei un Grande*. Sei determinato, spigliato, sempre attivo e io non posso fare altro che prenderti come esempio. Conserverò sempre un bellissimo ricordo del nostro viaggio negli States e spero che quello sia stato solo il primo di una lunga serie.

Concludo con un unico grande ringraziamento per tutti gli amici, i fotografi, quelli del calcetto, le persone che mi sono state vicino e che non ho ringraziato singolarmente non perché meno importanti ma perché conosciute da poco o con le quali non ho dei contatti frequenti. Nonostante questo avete contribuito tutti a farmi diventare la persona che sono oggi, con le sue esperienze e con la fortuna di poter dire di avervi incontrati.

A

MULTIMODAL MAPS

In this appendix we report the maps explored during the multimodal experiments with the exploration paths. Maps are divided by subjects.

For each subjects, the first 3 figures shows the results of exp 0 with the three conditions:

- Only TAMO
- Only Audio
- TAMO + Audio

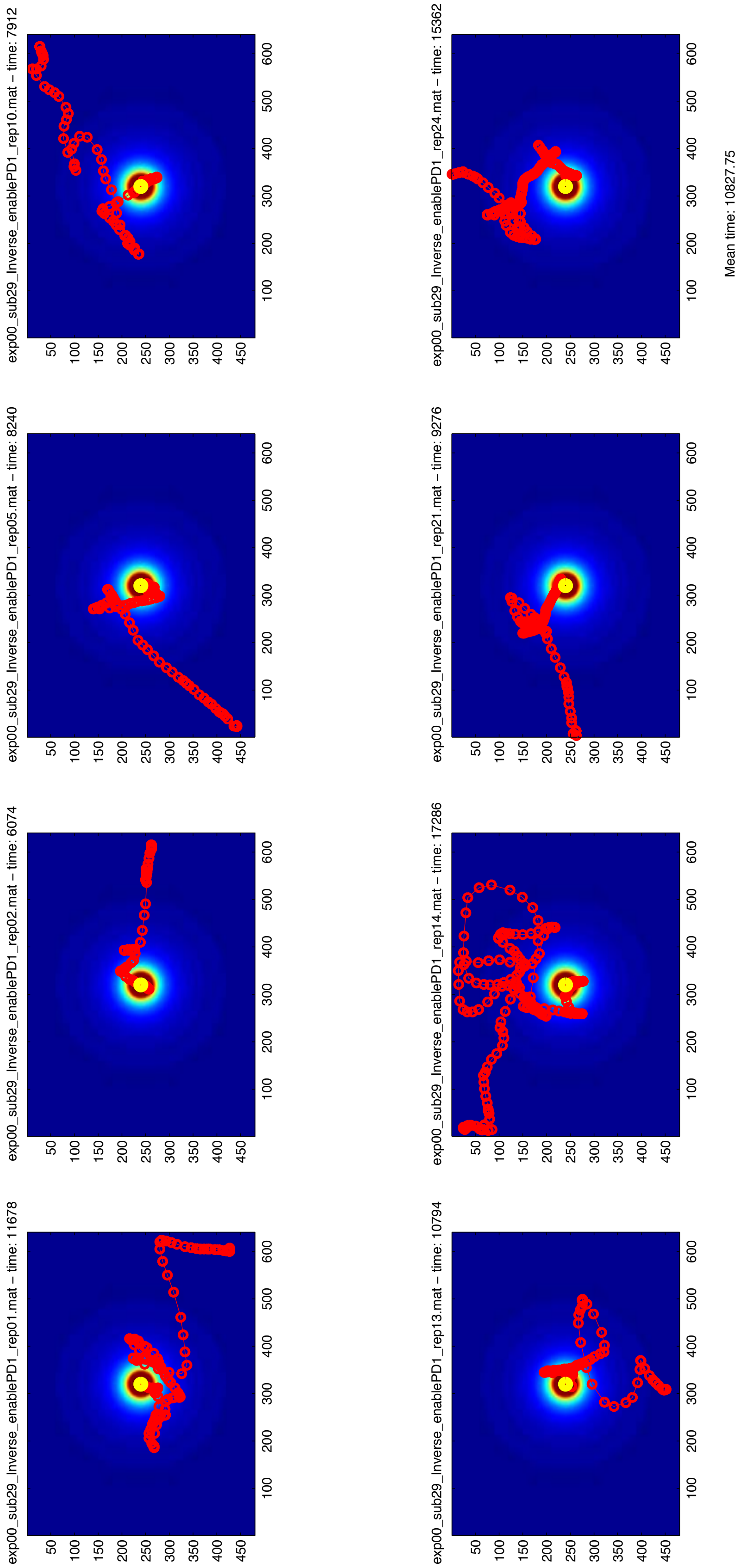
The 4th figure contains the results of exp 1. The first column refers to the exploration without audio, while the second column refers to the exploration with audio.

The last three figures of each subject refer to exp 2. The first one refers to the experiment with only the TAMO. The second one refers to the experiment both TAMO and audio 2D, while the third one refers to the experiment with both TAMO and audio 3D. The chosen position for the cube is shown on white with the relative localization error in pixel.

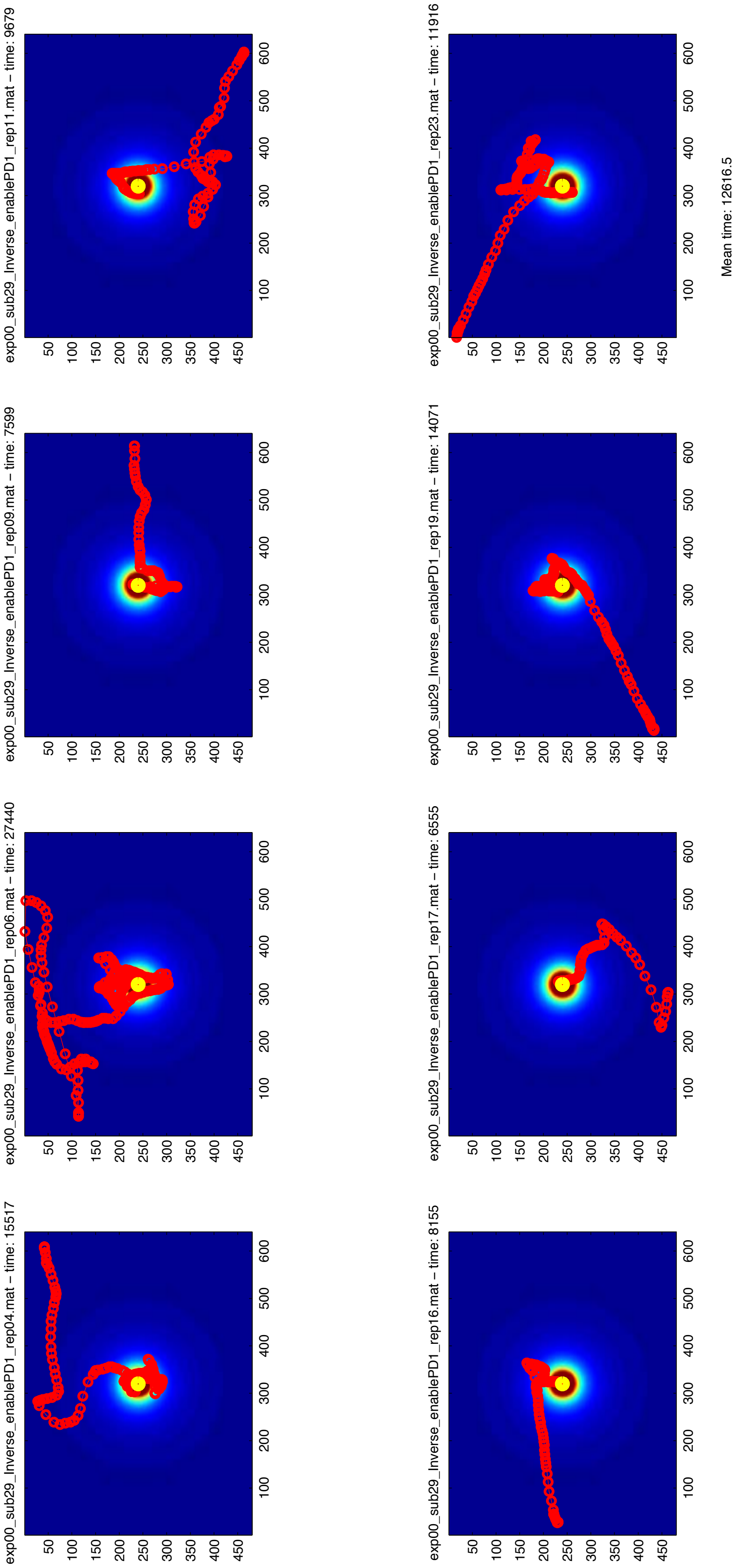
For all figures, titles allow to identify the particular trial performed. They also reported the total execution time.

A.1 SUBJECT 29

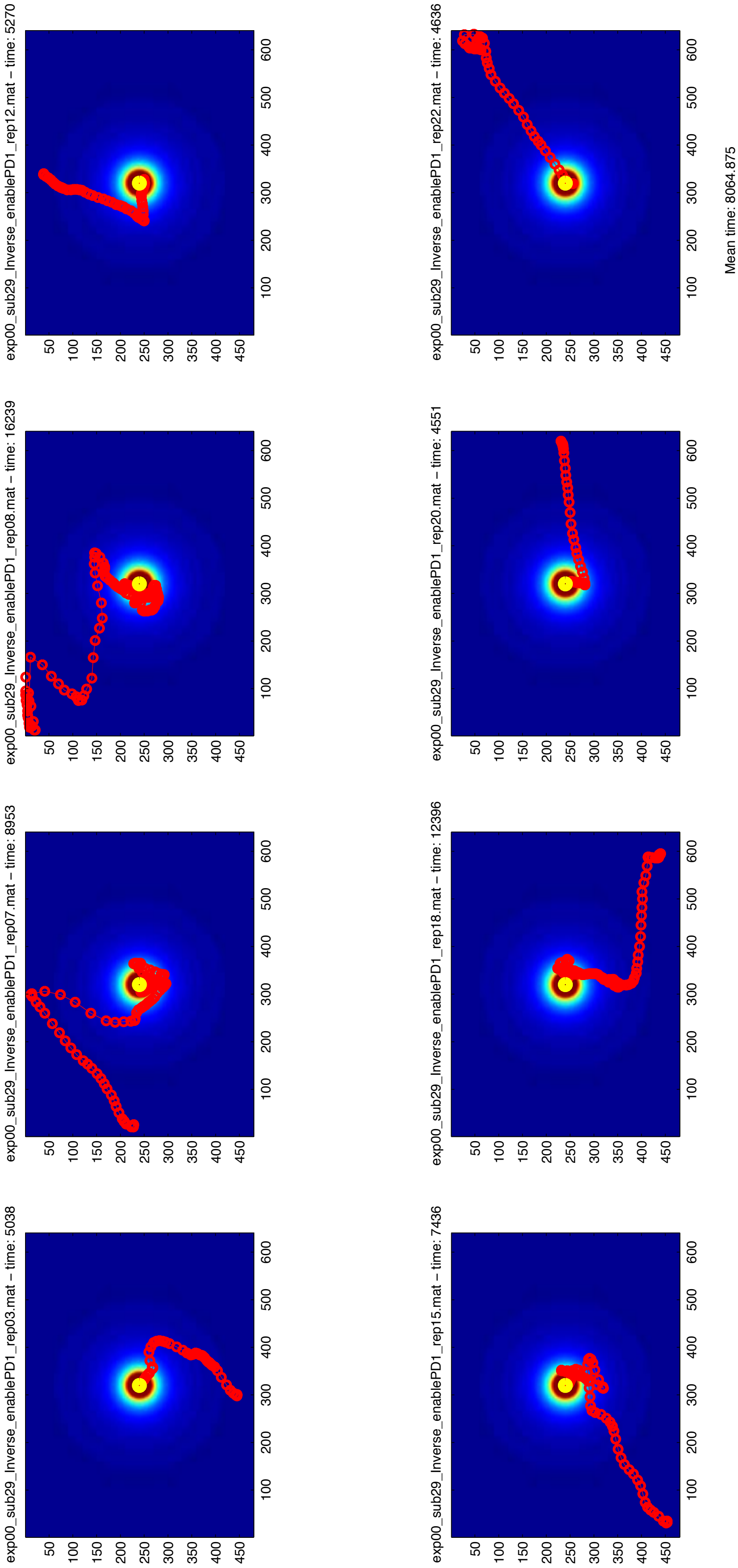
EXP 0 - ONLY TAMO



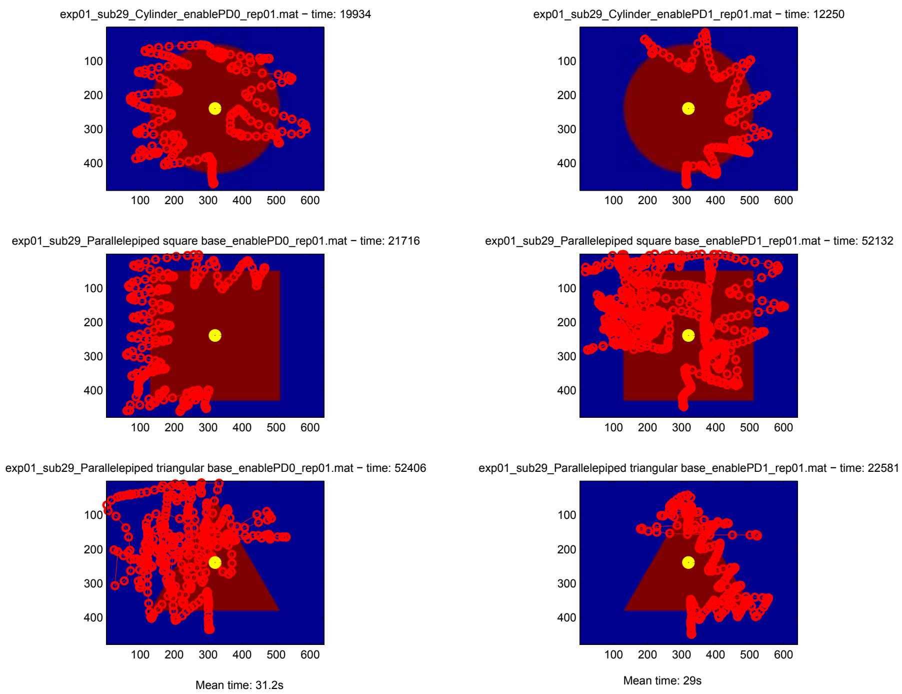
EXP 0 - ONLY AUDIO



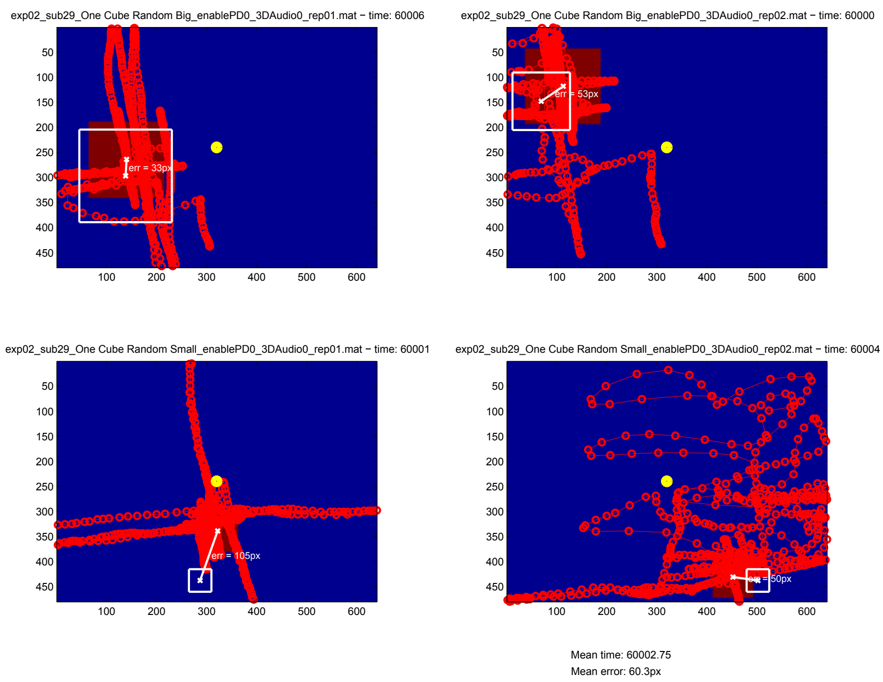
EXP 0 - TAMO AND AUDIO



EXP 1 - NO AUDIO AGAINST AUDIO

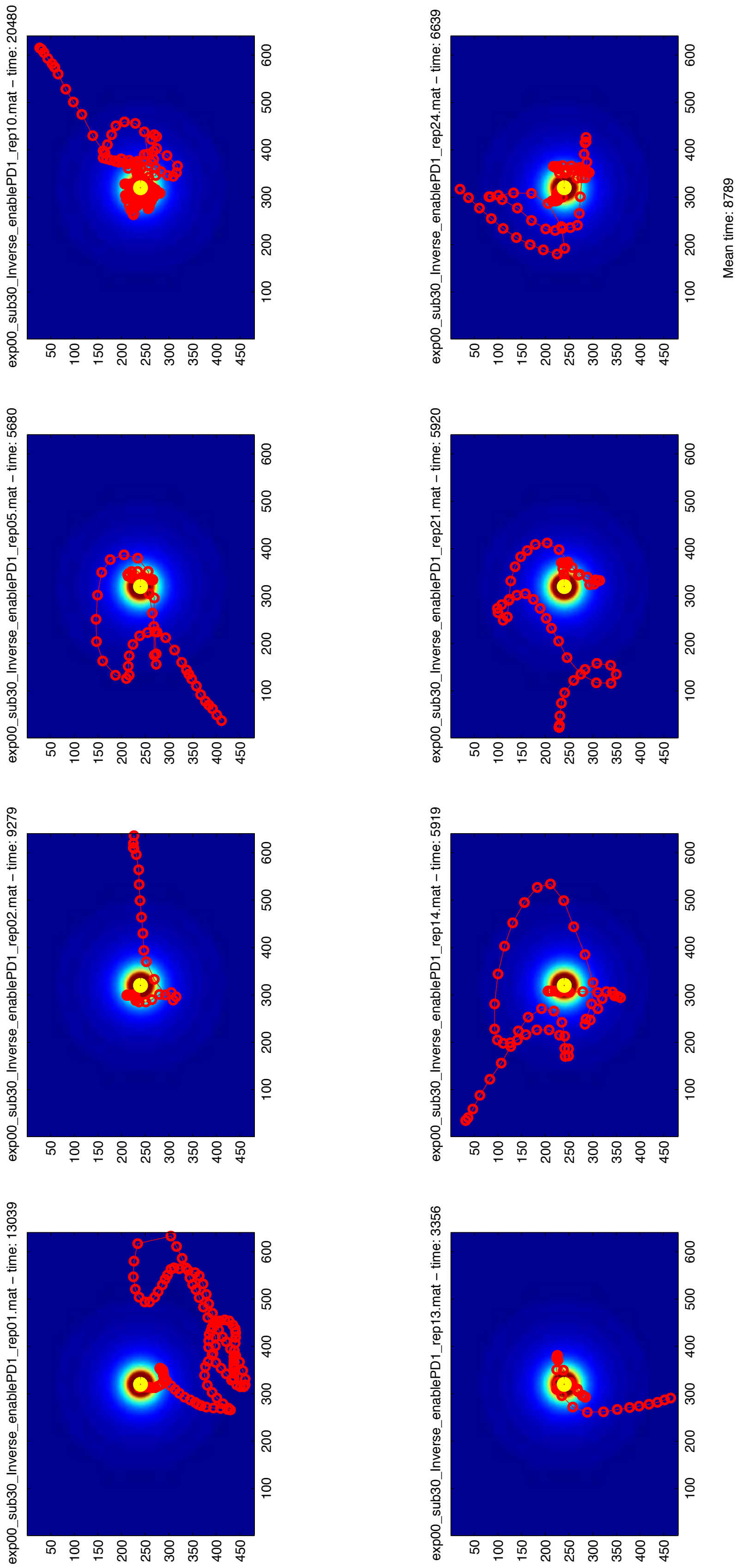


EXP 2 - ONLY TAMO

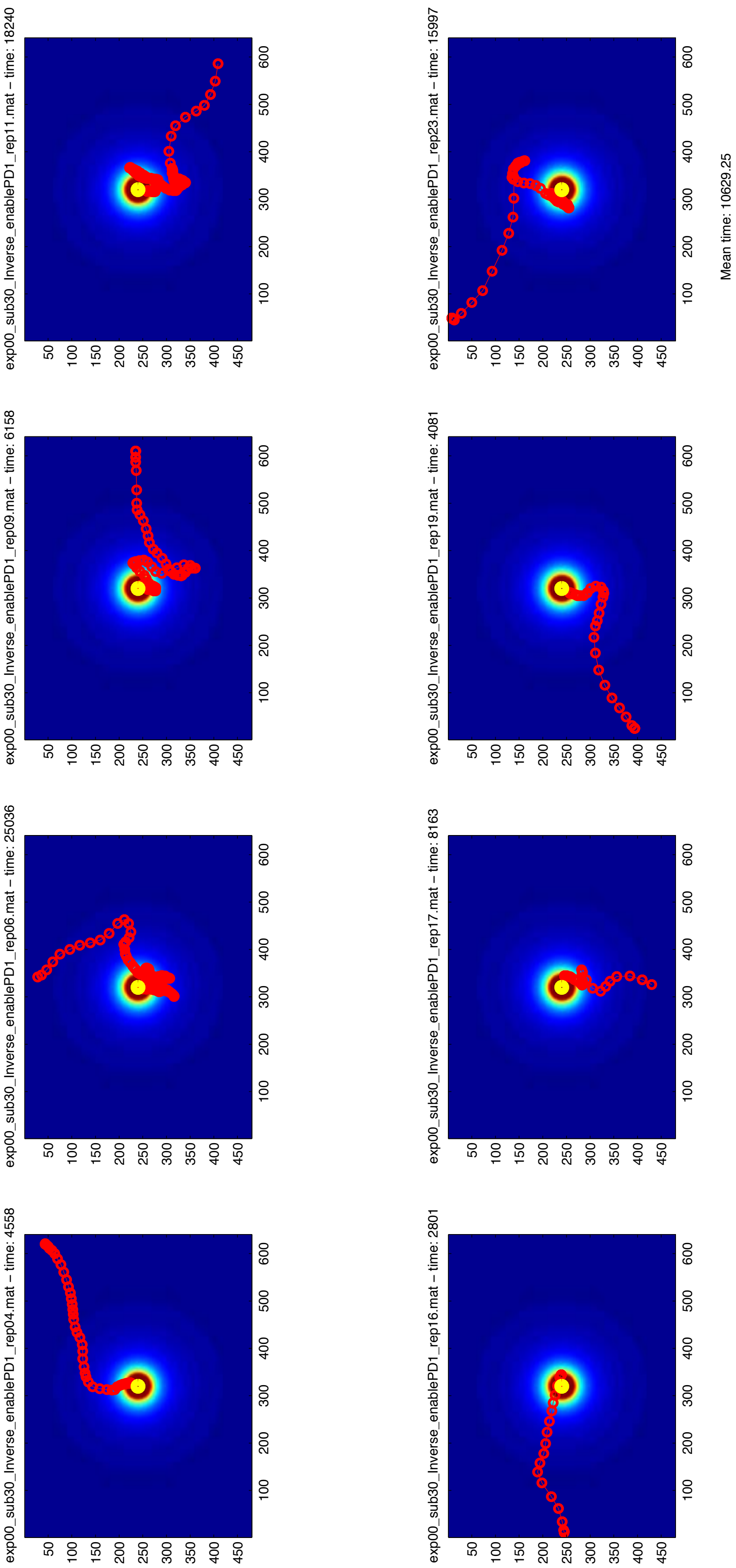


A.2 SUBJECT 30

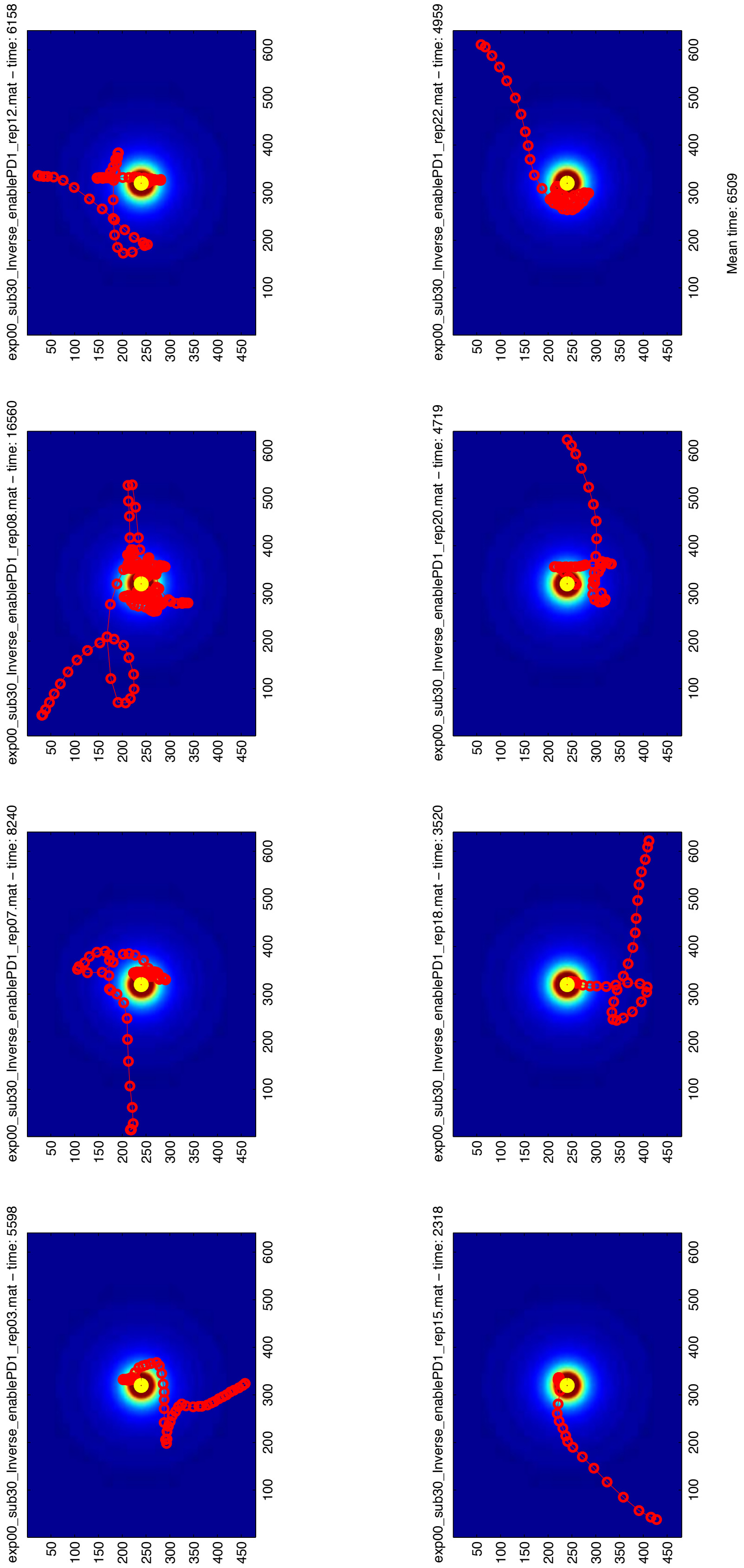
EXP 0 - ONLY TAMO



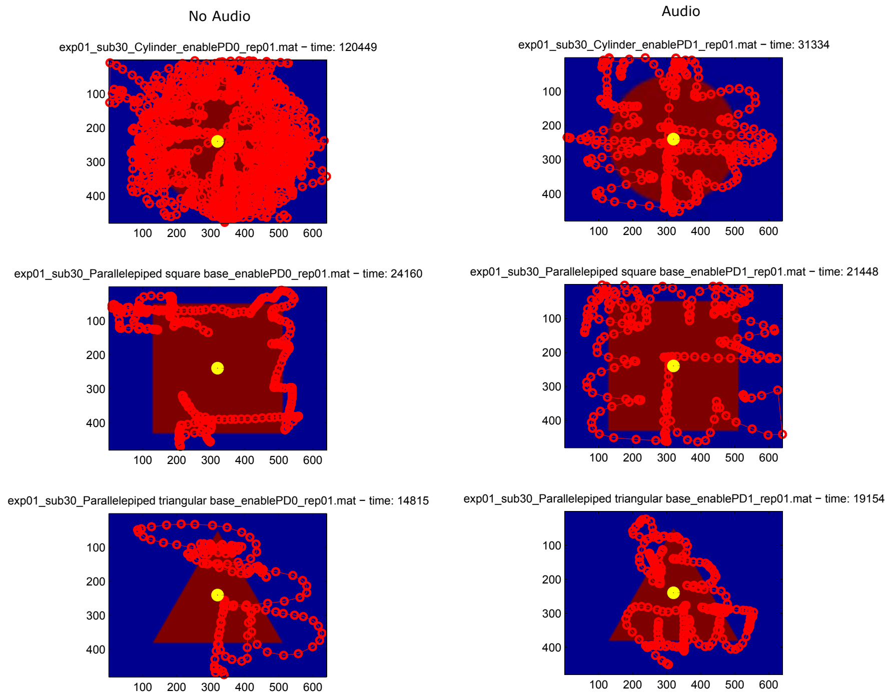
EXP 0 - ONLY AUDIO



EXP 0 - TAMO AND AUDIO



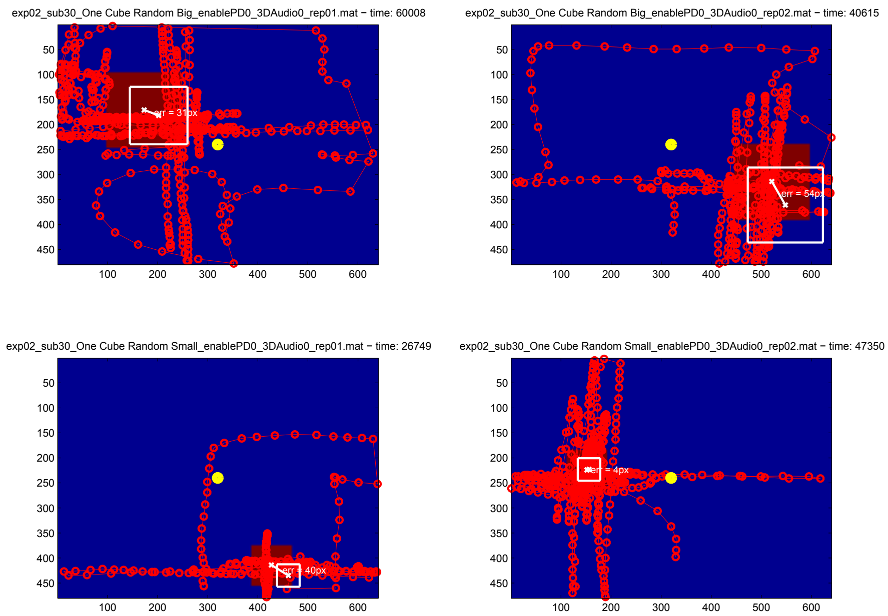
EXP 1 - NO AUDIO AGAINST AUDIO



Mean time: 53.0s

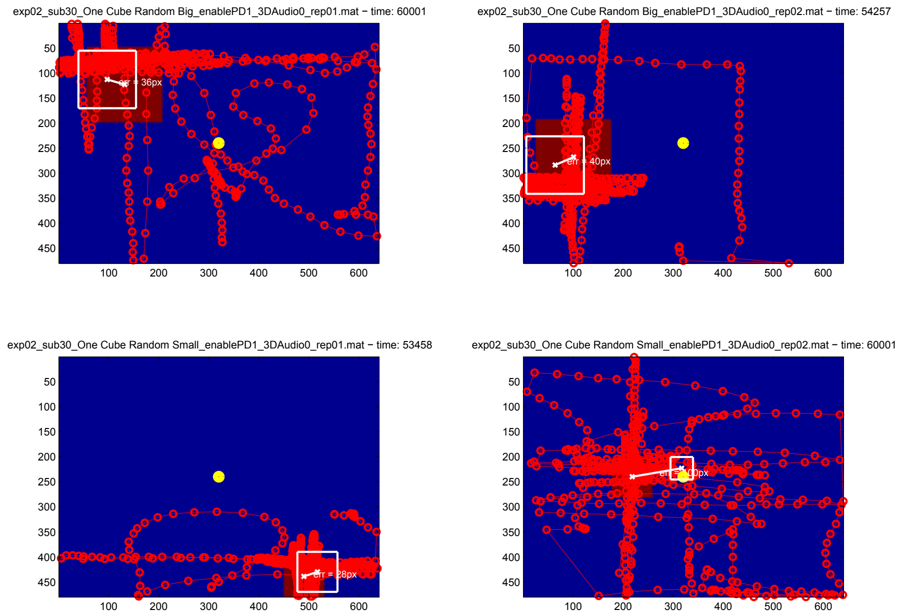
Mean time: 23.9

EXP 2 - ONLY TAMO



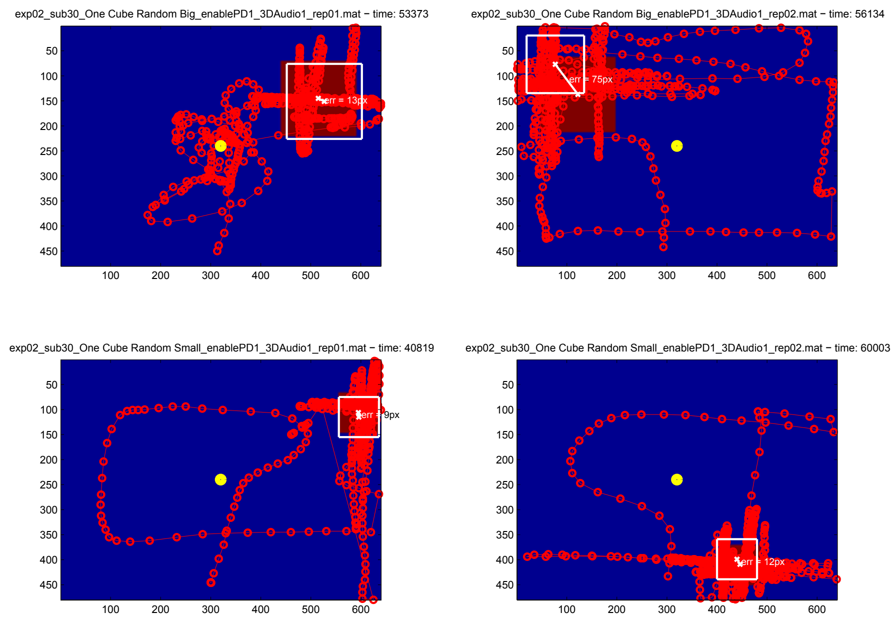
Mean time: 43680.5
Mean error: 32px

EXP 2 - TAMO AND AUDIO 2D



Mean time: 56929.25
Mean error: 51px

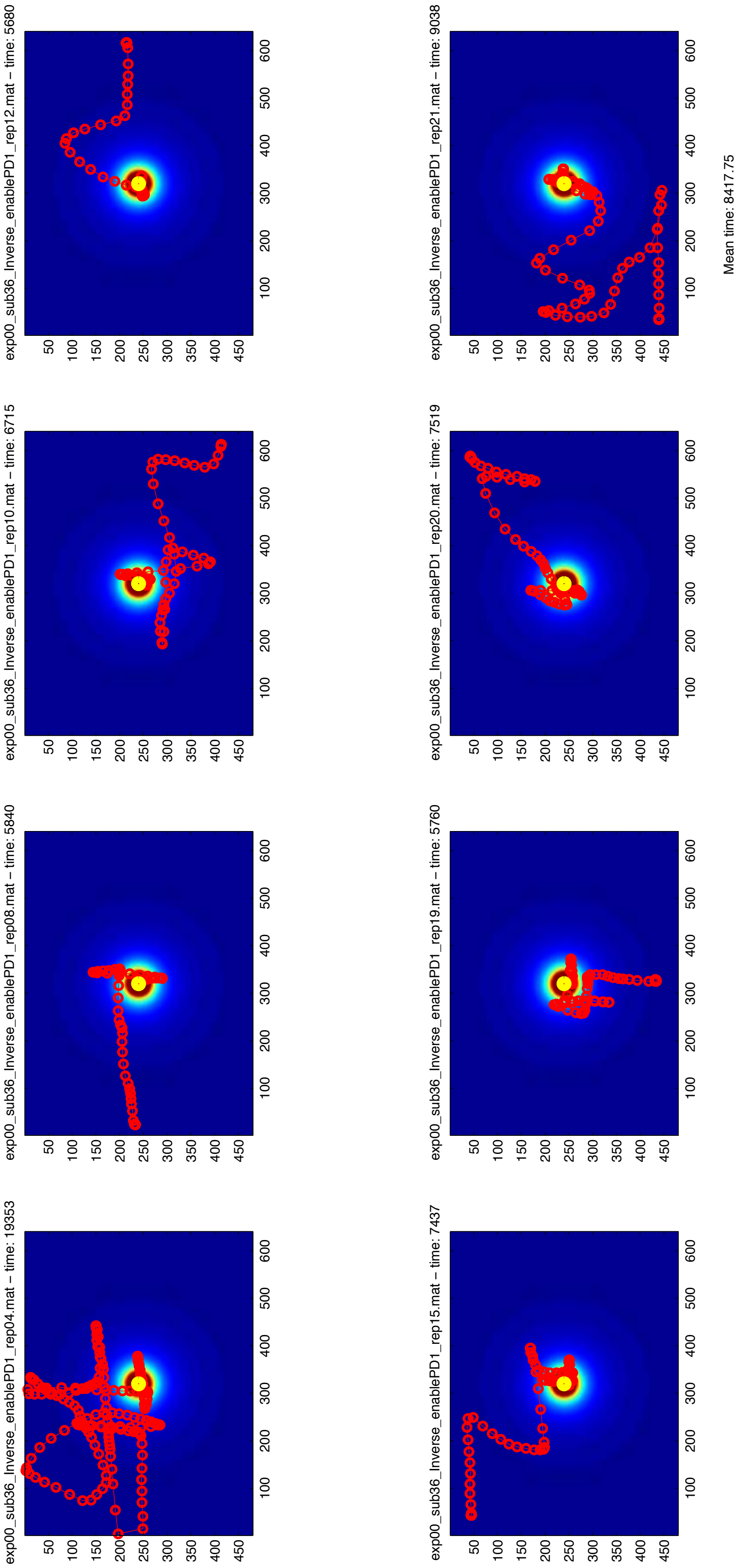
EXP 2 - TAMO AND AUDIO 3D



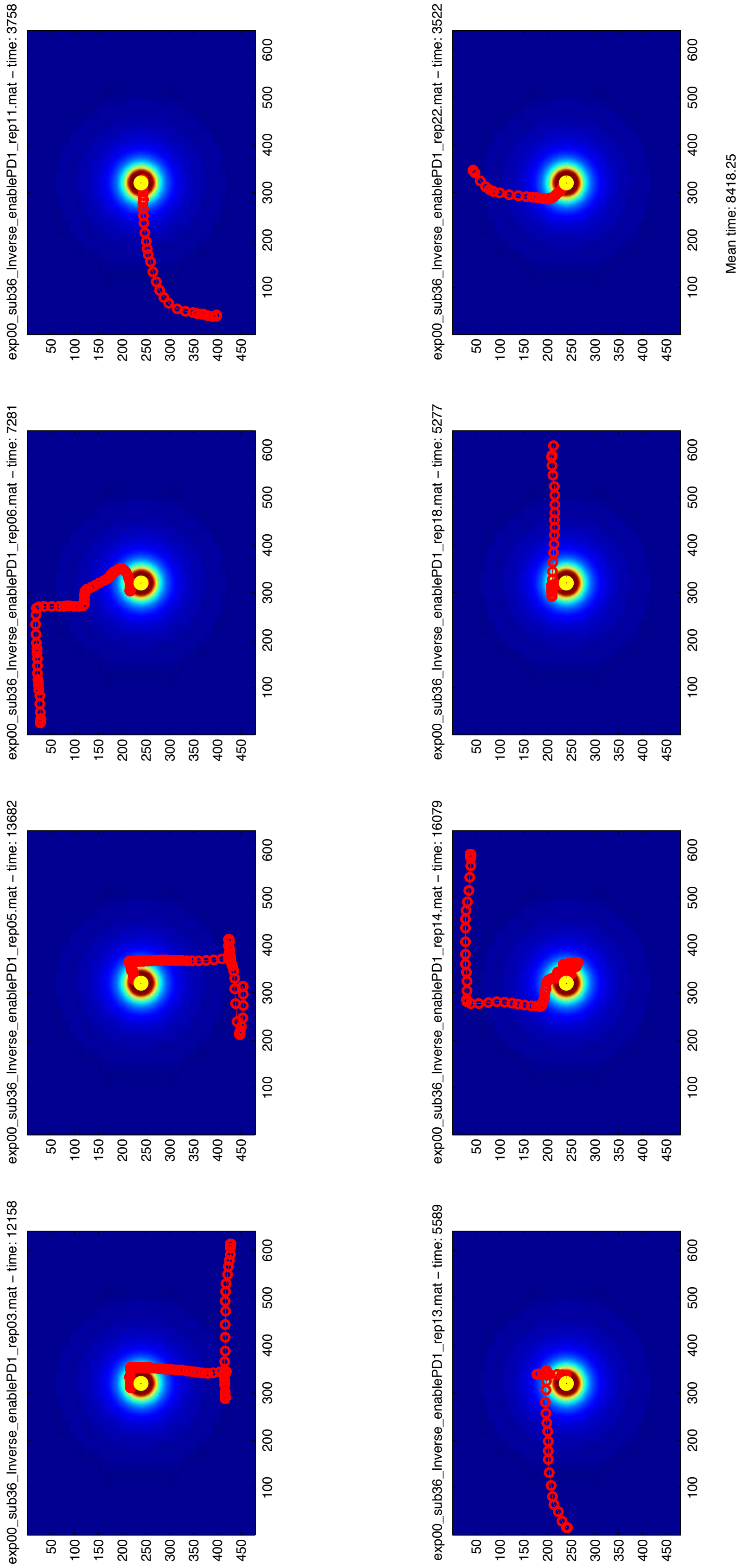
Mean time: 52582.25
Mean error: 27px

A.3 SUBJECT 36

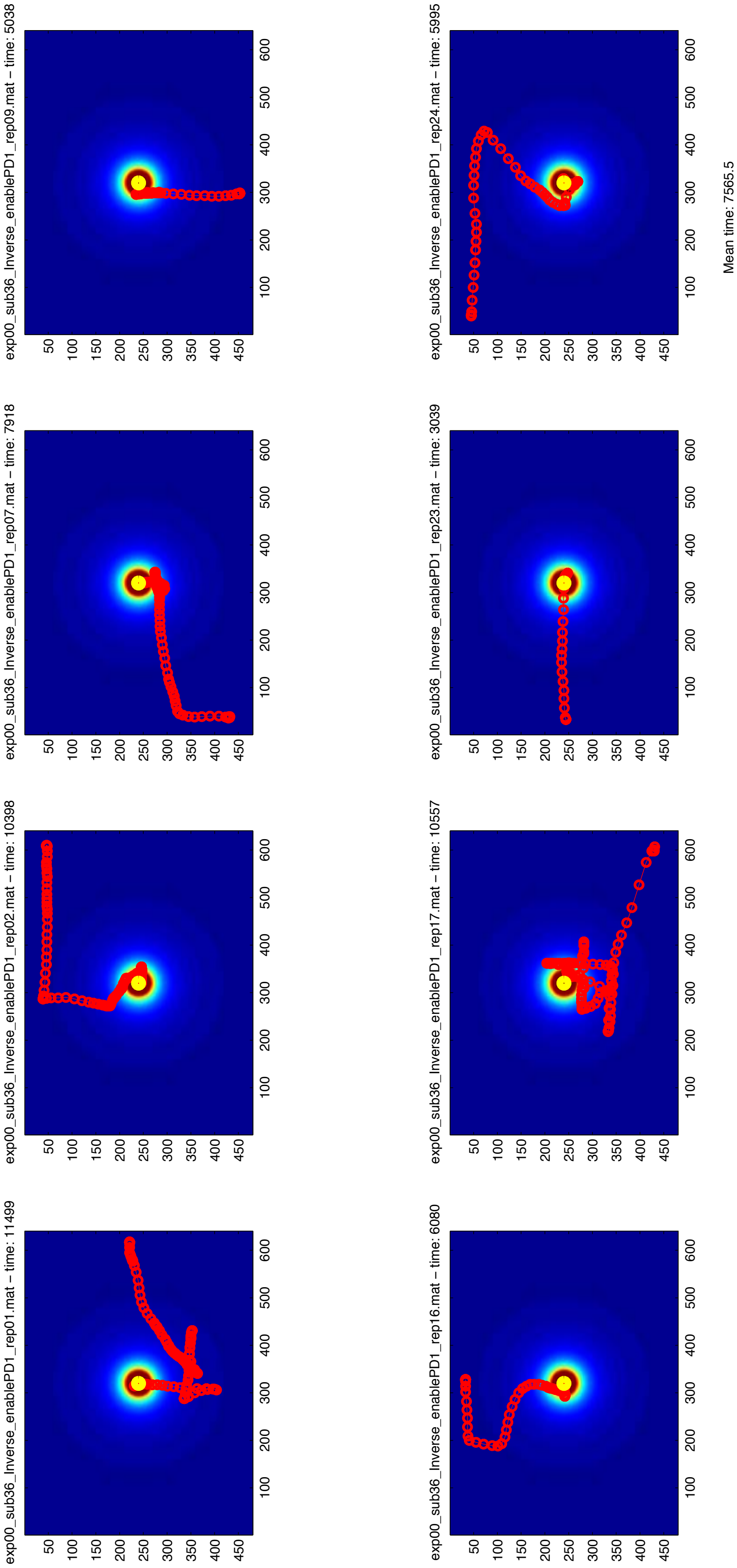
EXP 0 - ONLY TAMO



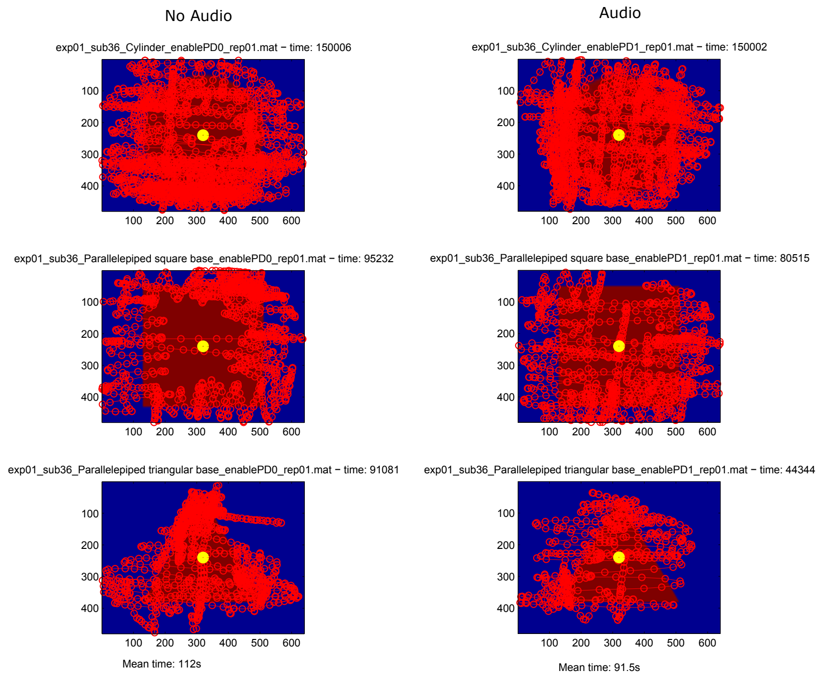
EXP 0 - ONLY AUDIO



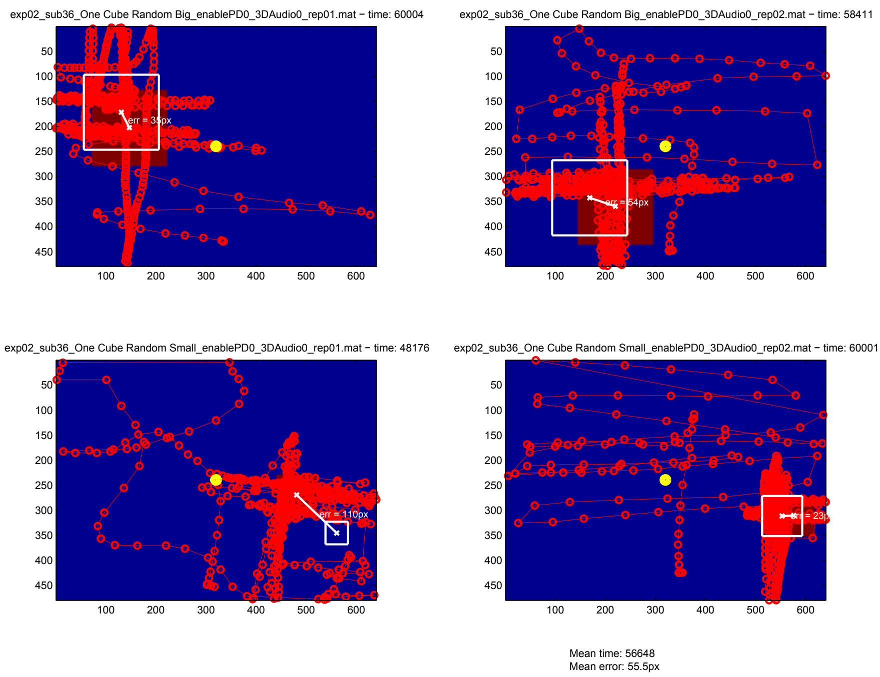
EXP 0 - TAMO AND AUDIO



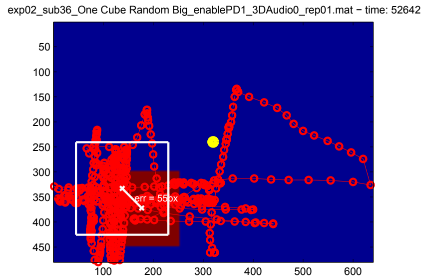
EXP 1 - NO AUDIO AGAINST AUDIO

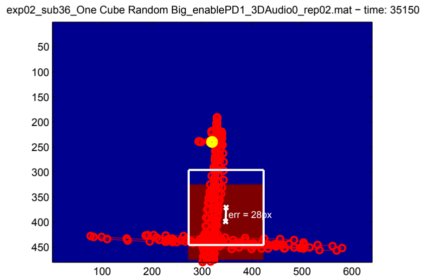


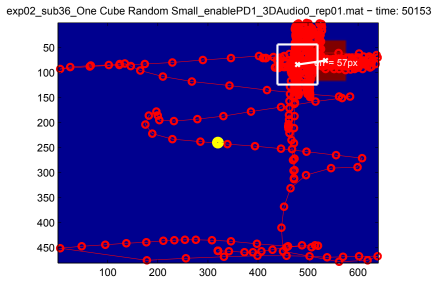
EXP 2 - ONLY TAMO

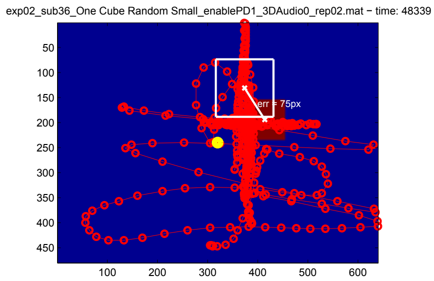


EXP 2 - TAMO AND AUDIO 2D



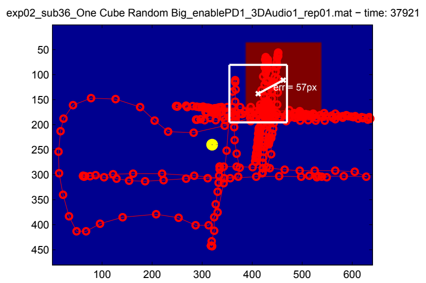


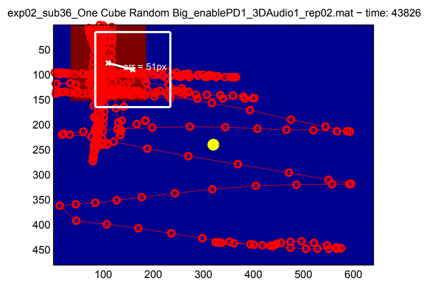


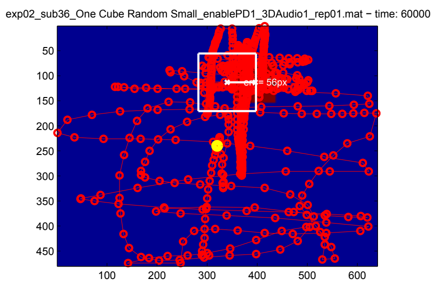


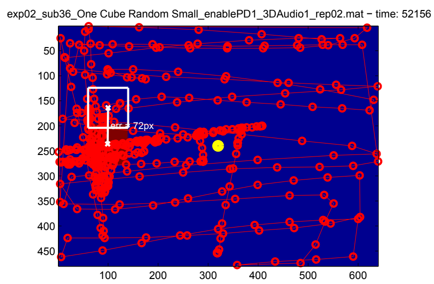
Mean time: 46571
Mean error: 54px

EXP 2 - TAMO AND AUDIO 3D





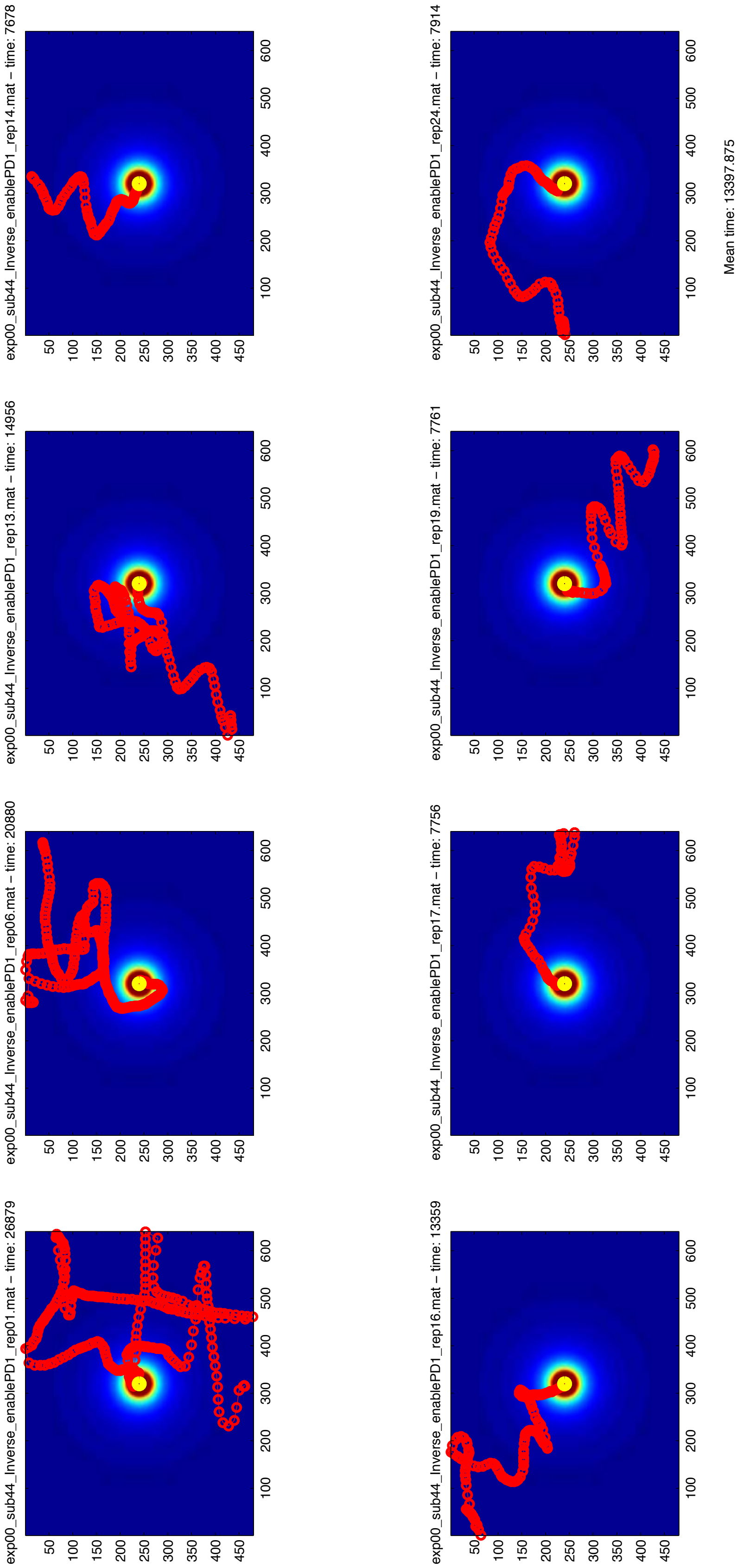




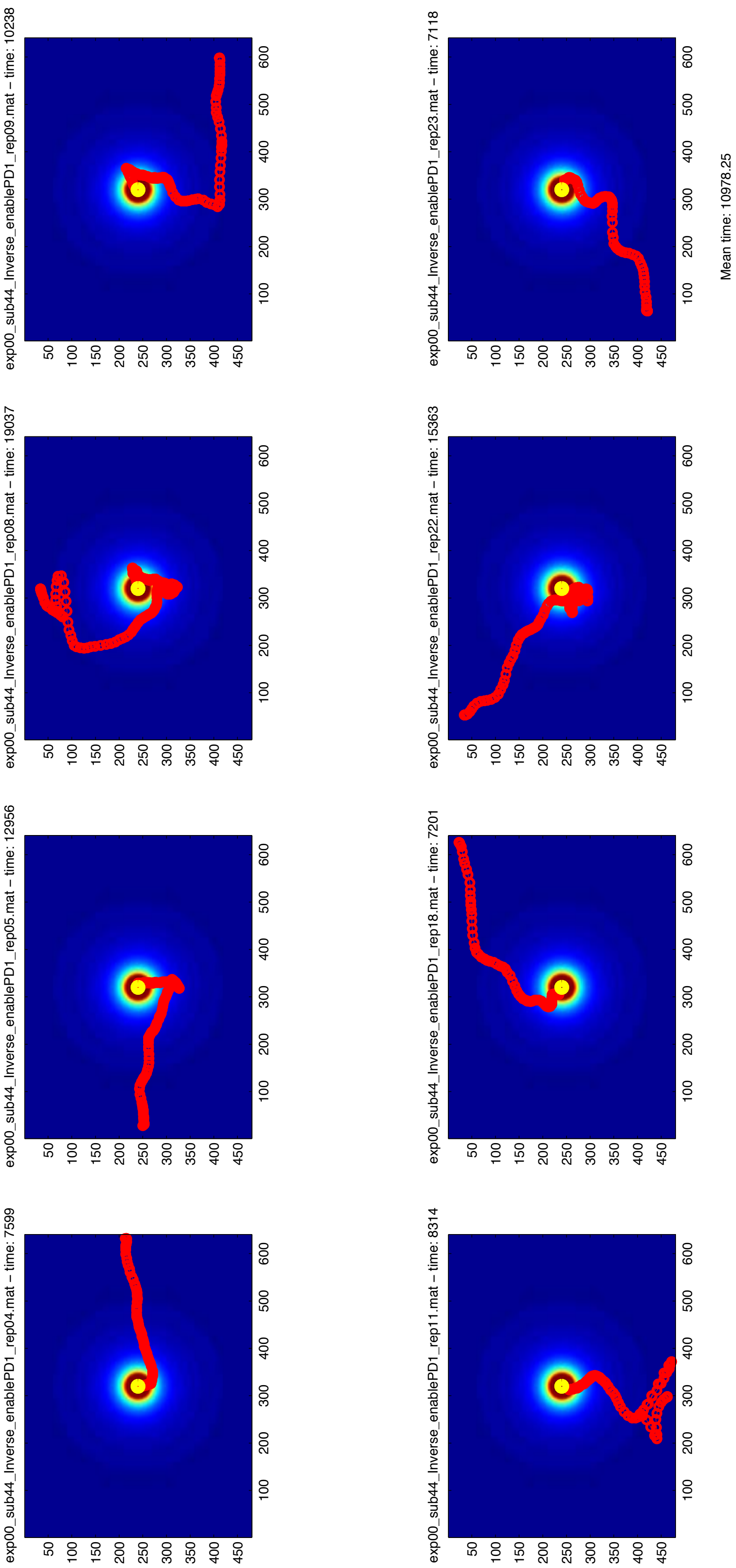
Mean time: 48475.75
Mean error: 59px

A.4 SUBJECT 44

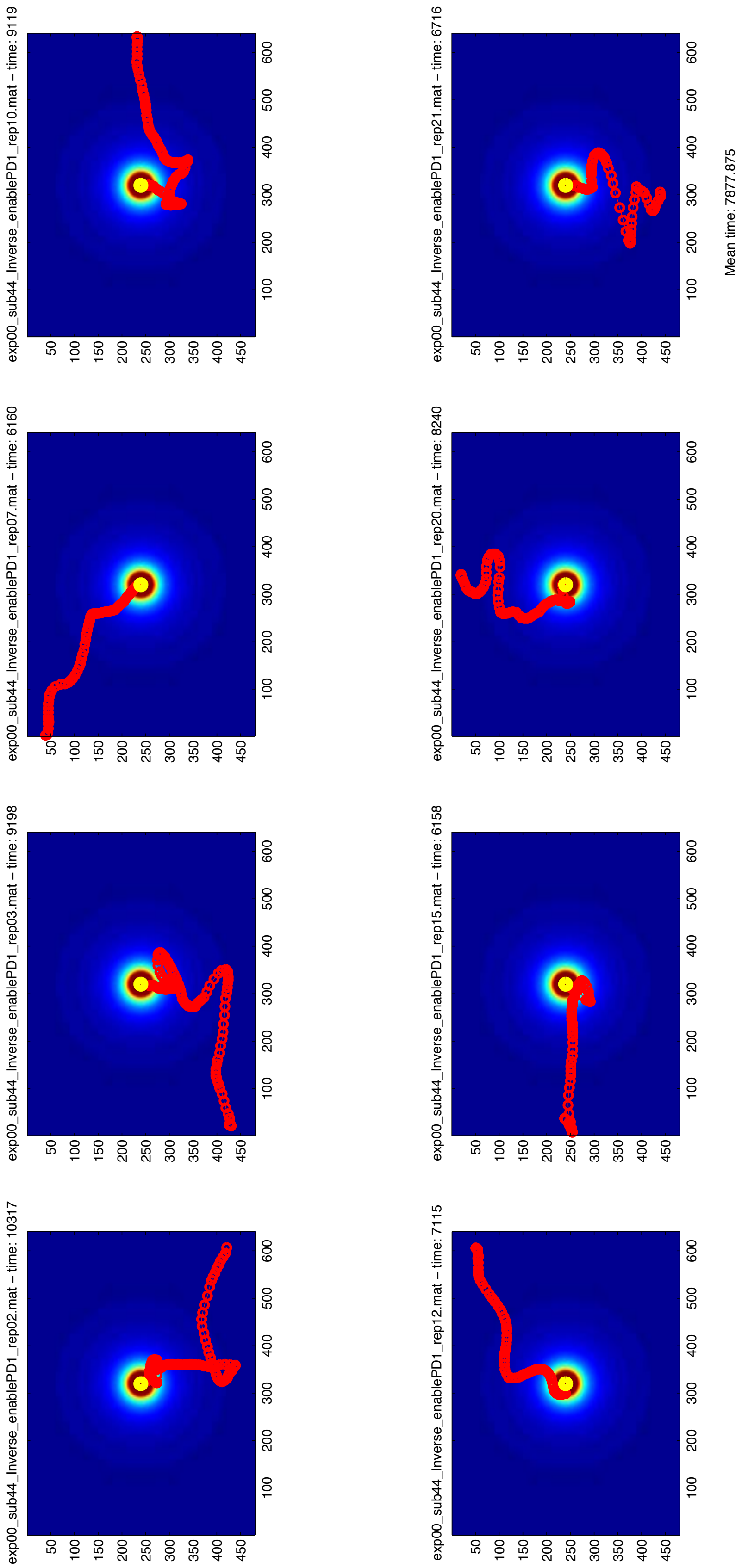
EXP 0 - ONLY TAMO



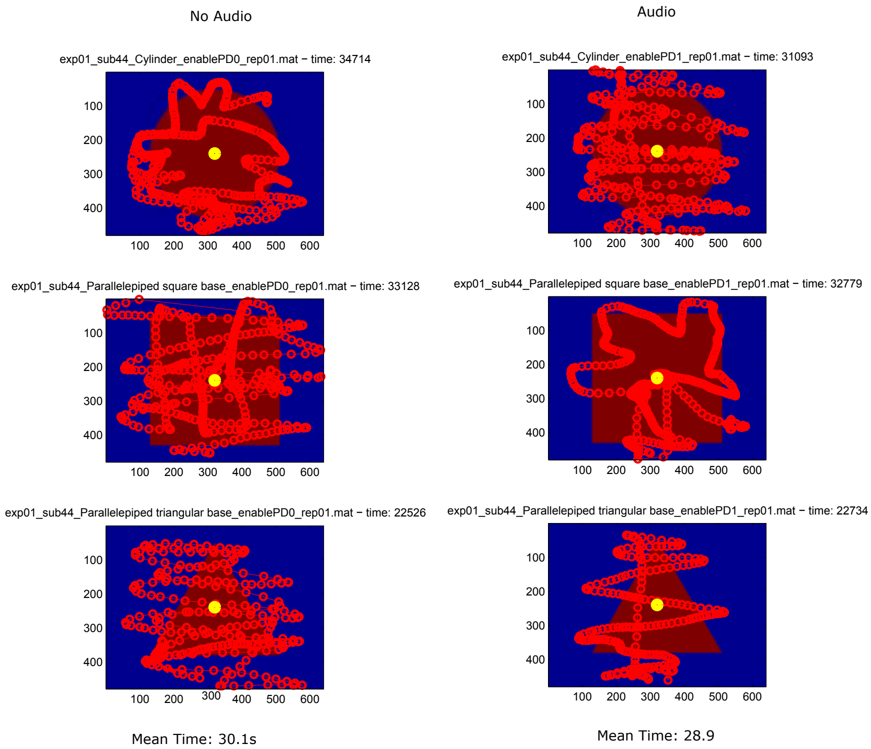
EXP 0 - ONLY AUDIO



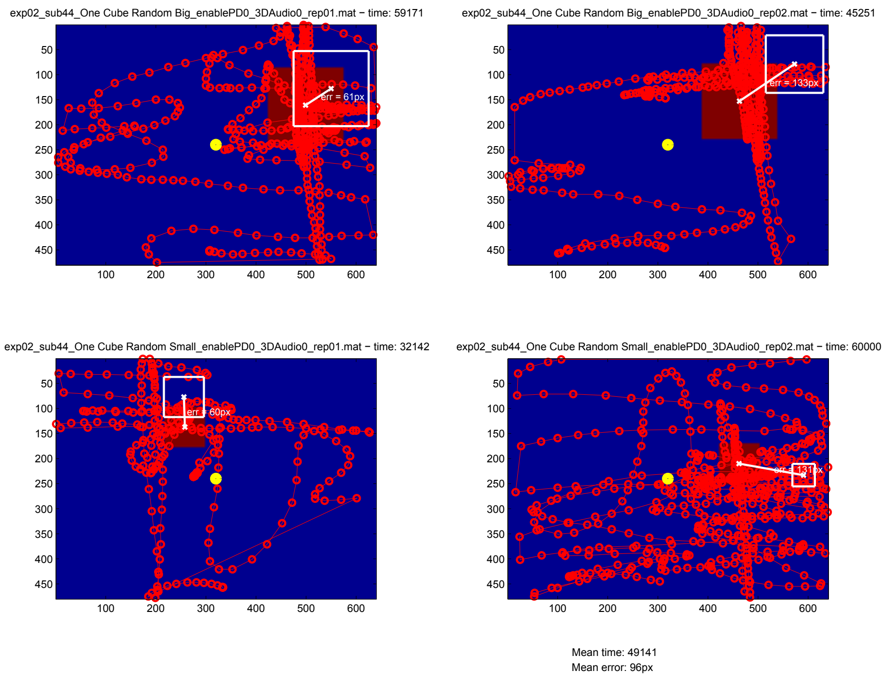
EXP 0 - TAMO AND AUDIO



EXP 1 - NO AUDIO AGAINST AUDIO

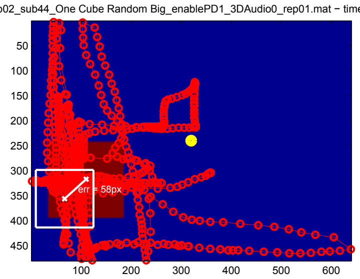


EXP 2 - ONLY TAMO

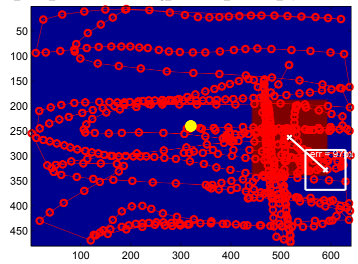


EXP 2 - TAMO AND AUDIO 2D

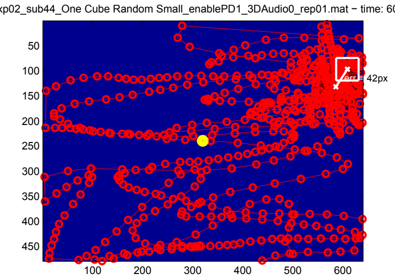
exp02_sub44_One Cube Random Big_enablePD1_3DAudio0_rep01.mat − time: 60004



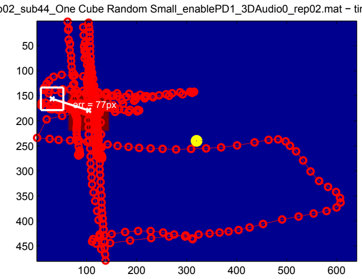
exp02_sub44_One Cube Random Big_enablePD1_3DAudio0_rep02.mat − time: 60001



exp02_sub44_One Cube Random Small_enablePD1_3DAudio0_rep01.mat − time: 60001



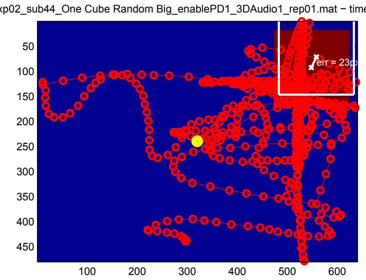
exp02_sub44_One Cube Random Small_enablePD1_3DAudio0_rep02.mat − time: 38387



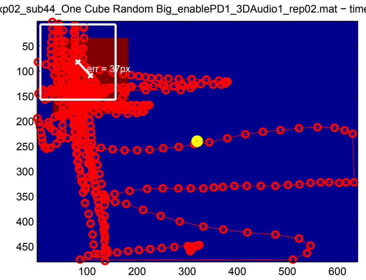
Mean time: 54598.25
Mean error: 69px

EXP 2 - TAMO AND AUDIO 3D

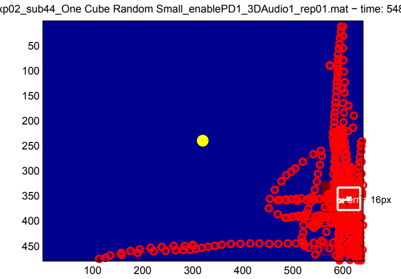
exp02_sub44_One Cube Random Big_enablePD1_3DAudio1_rep01.mat − time: 60001



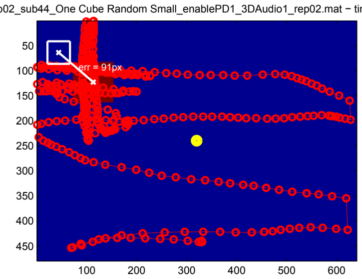
exp02_sub44_One Cube Random Big_enablePD1_3DAudio1_rep02.mat − time: 59546



exp02_sub44_One Cube Random Small_enablePD1_3DAudio1_rep01.mat − time: 54806



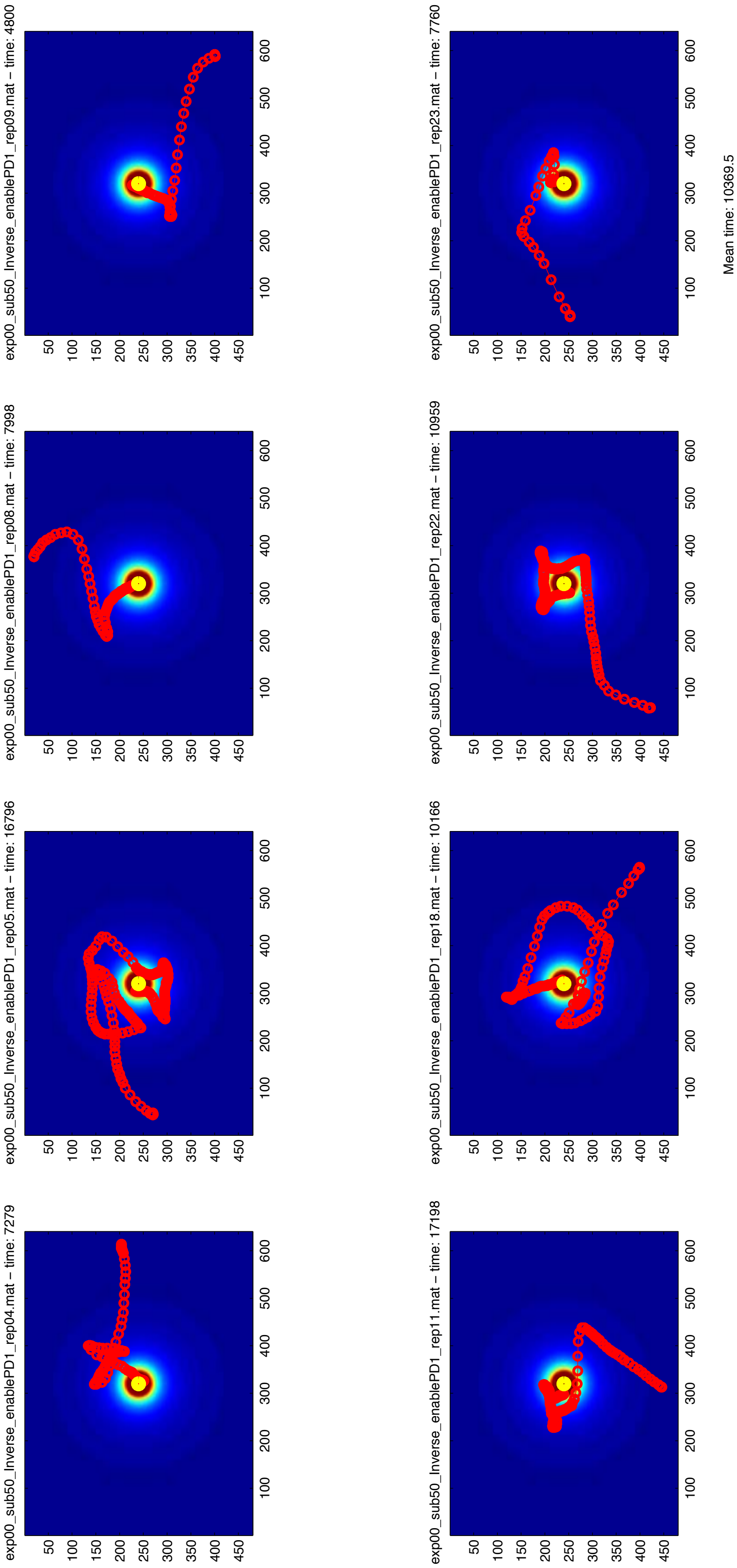
exp02_sub44_One Cube Random Small_enablePD1_3DAudio1_rep02.mat − time: 43230



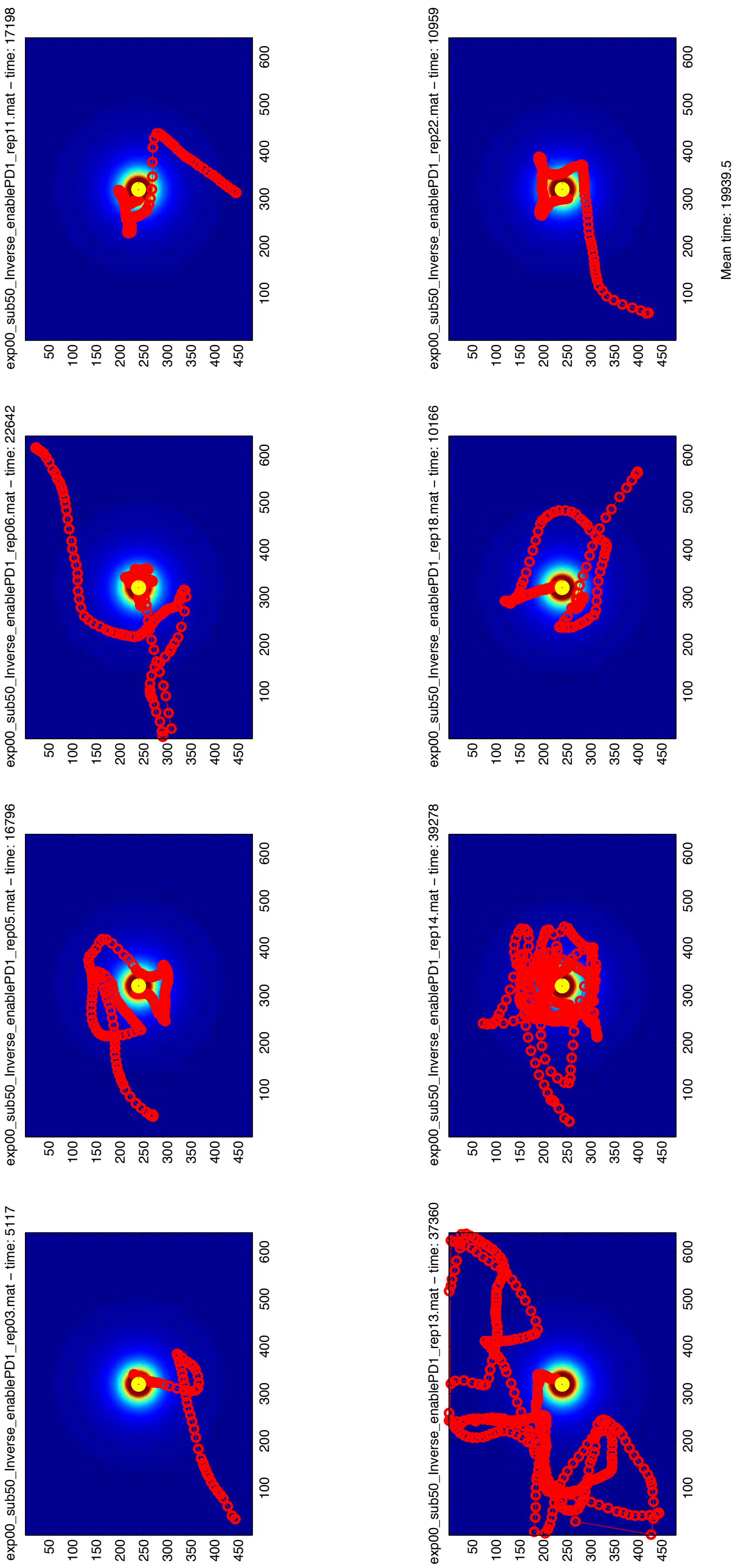
Mean time: 54395.75
Mean error: 41.8px

A.5 SUBJECT 50

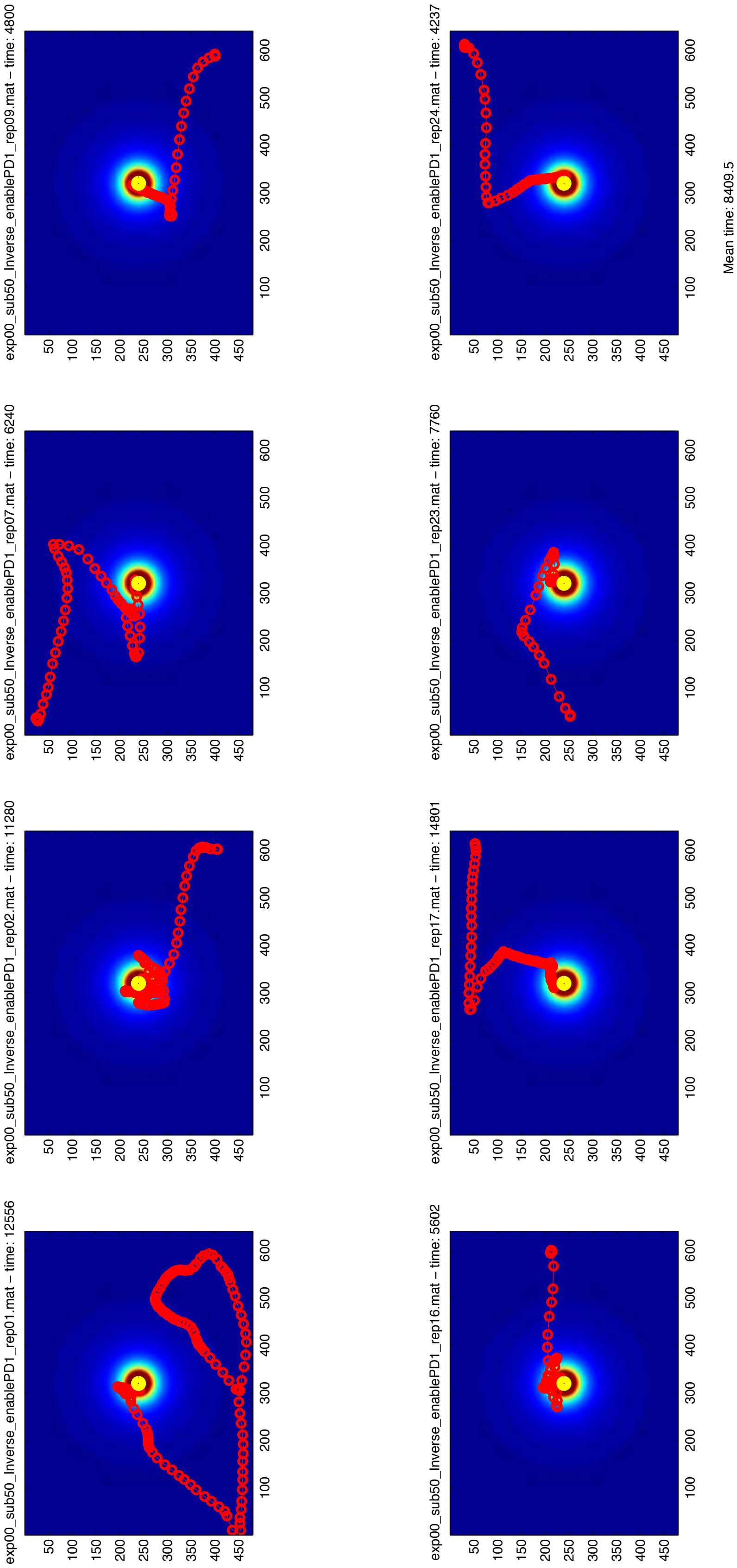
EXP 0 - ONLY TAMO



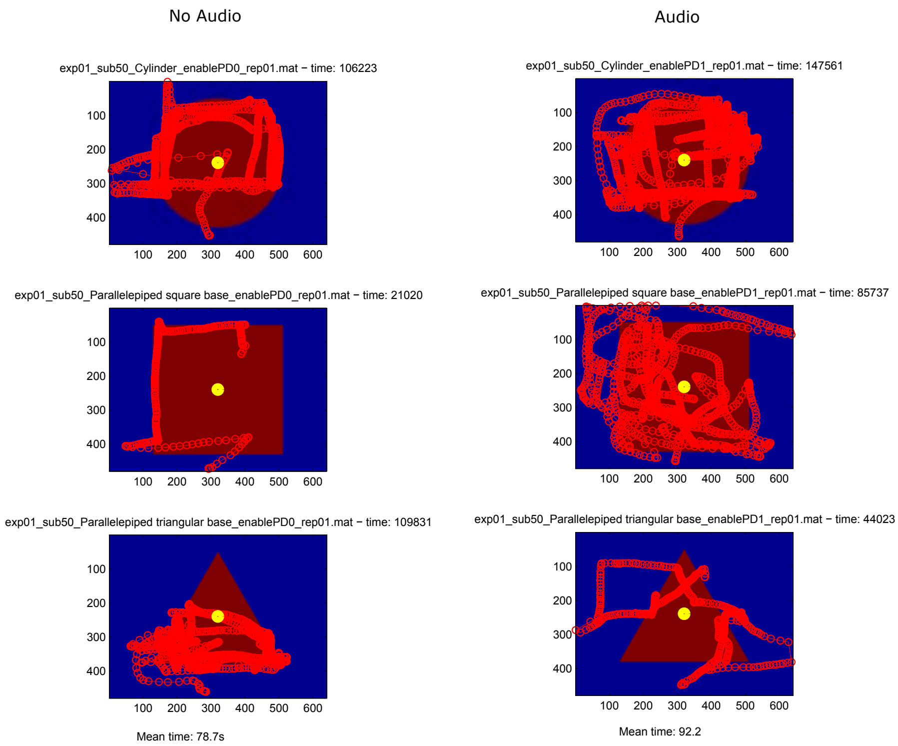
EXP 0 - ONLY AUDIO



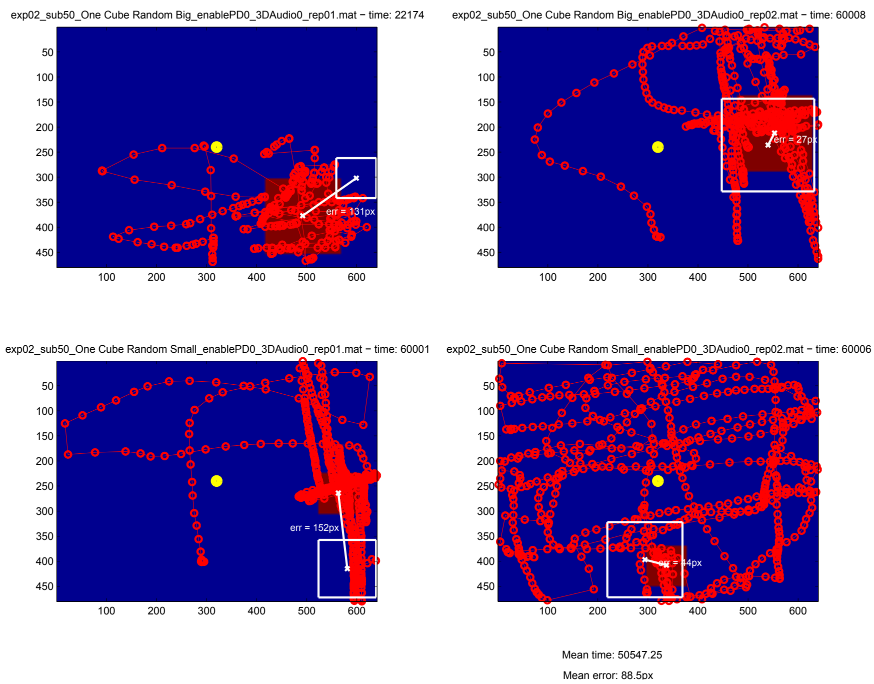
exp 0 - tamo and audio



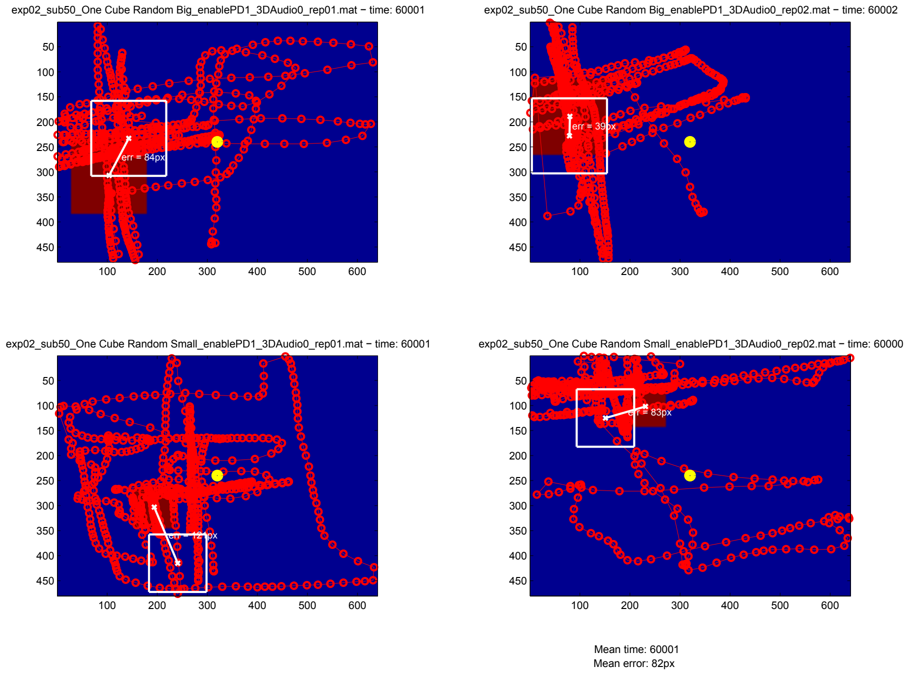
EXP 1 - NO AUDIO AGAINST AUDIO



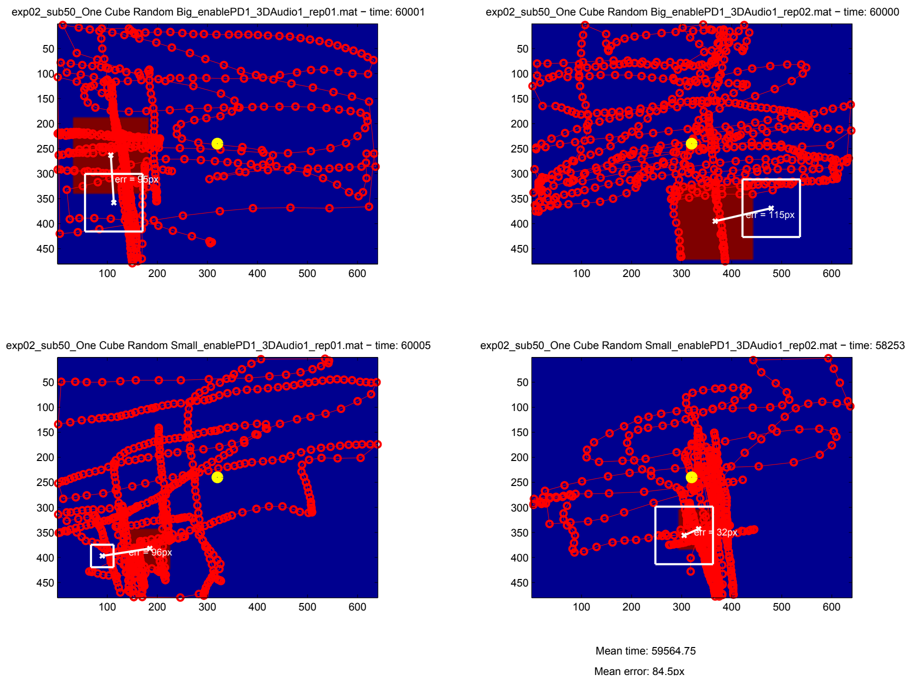
EXP 2 - ONLY TAMO



exp 2 - tamo and audio 2d

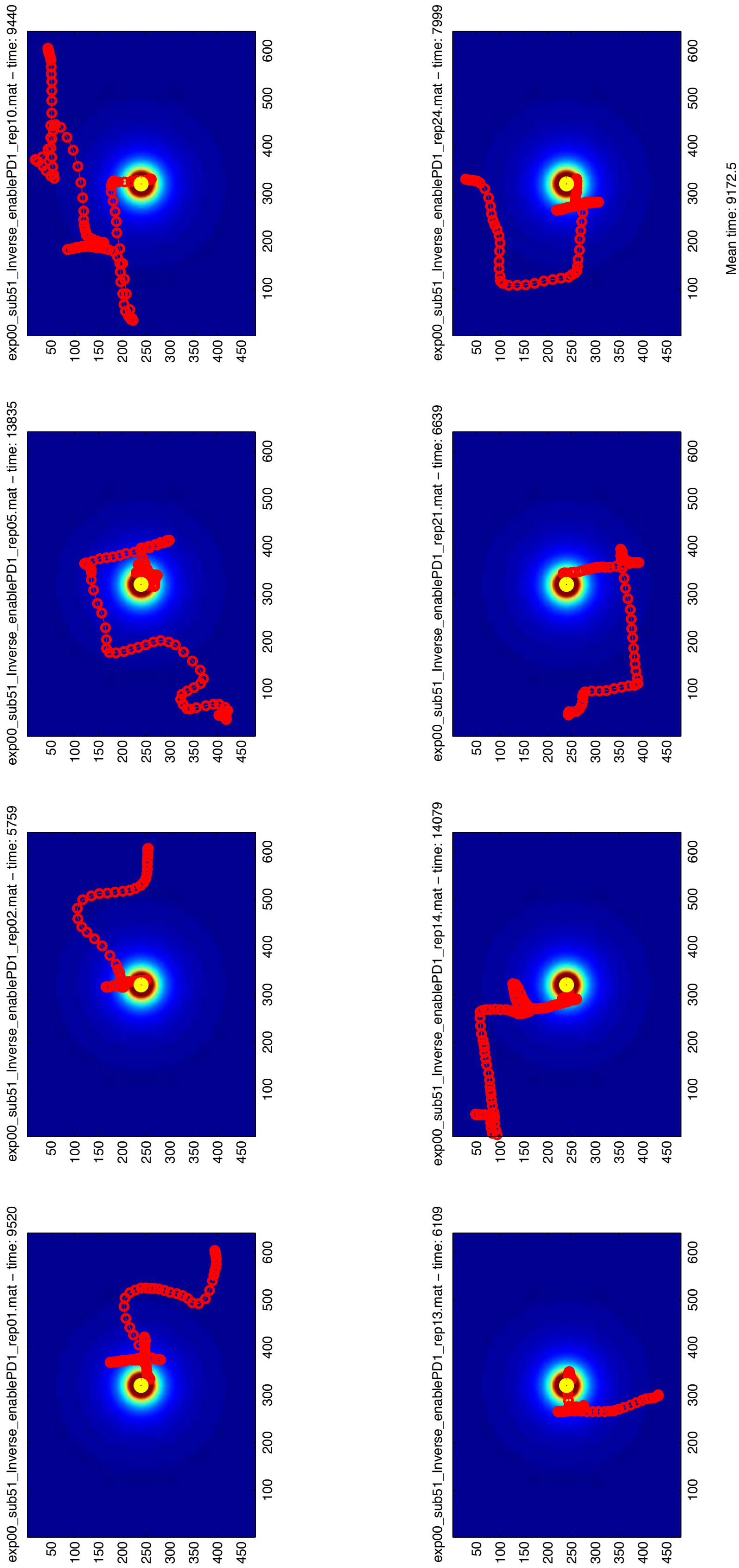


exp 2 - tamo and audio 3d

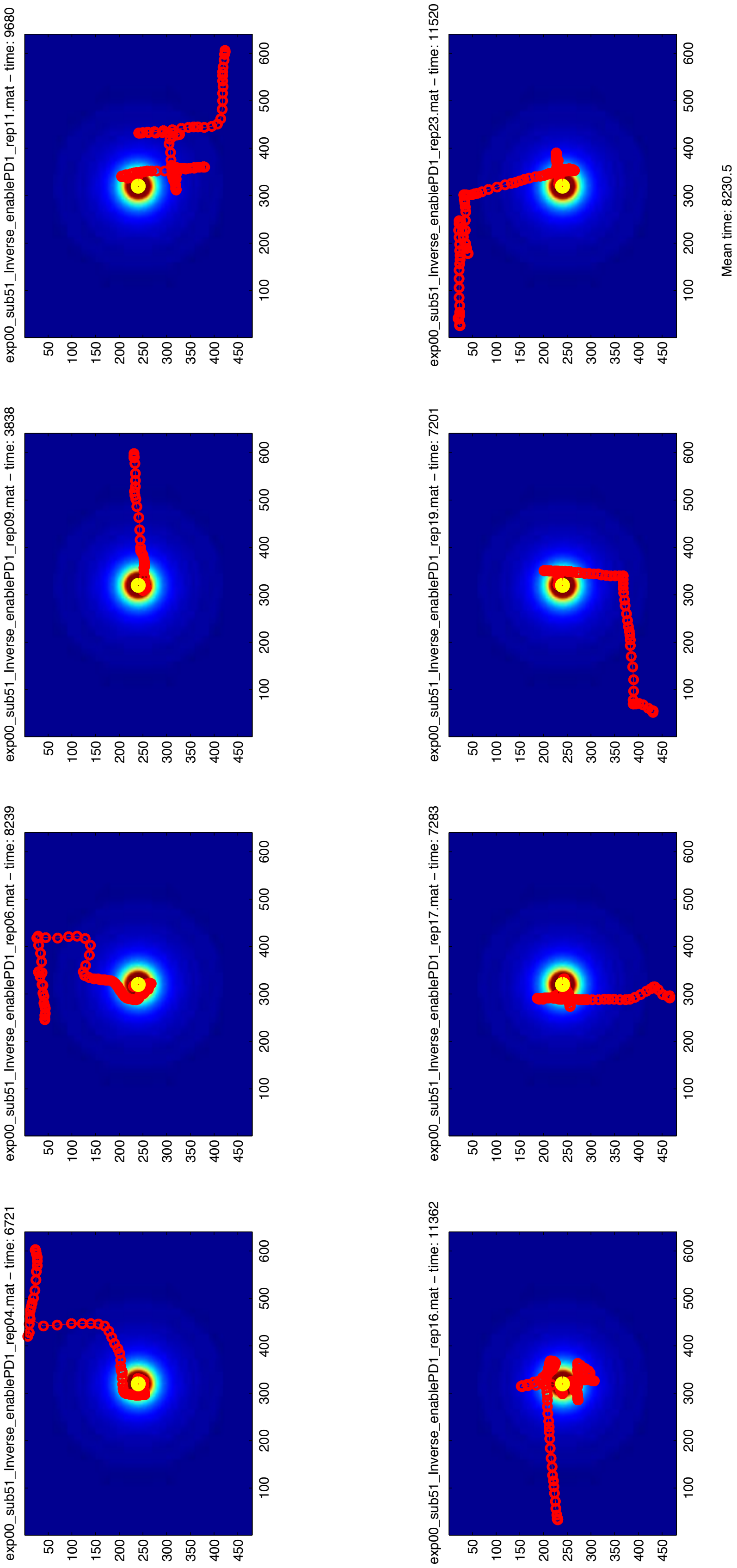


A.6 SUBJECT 51

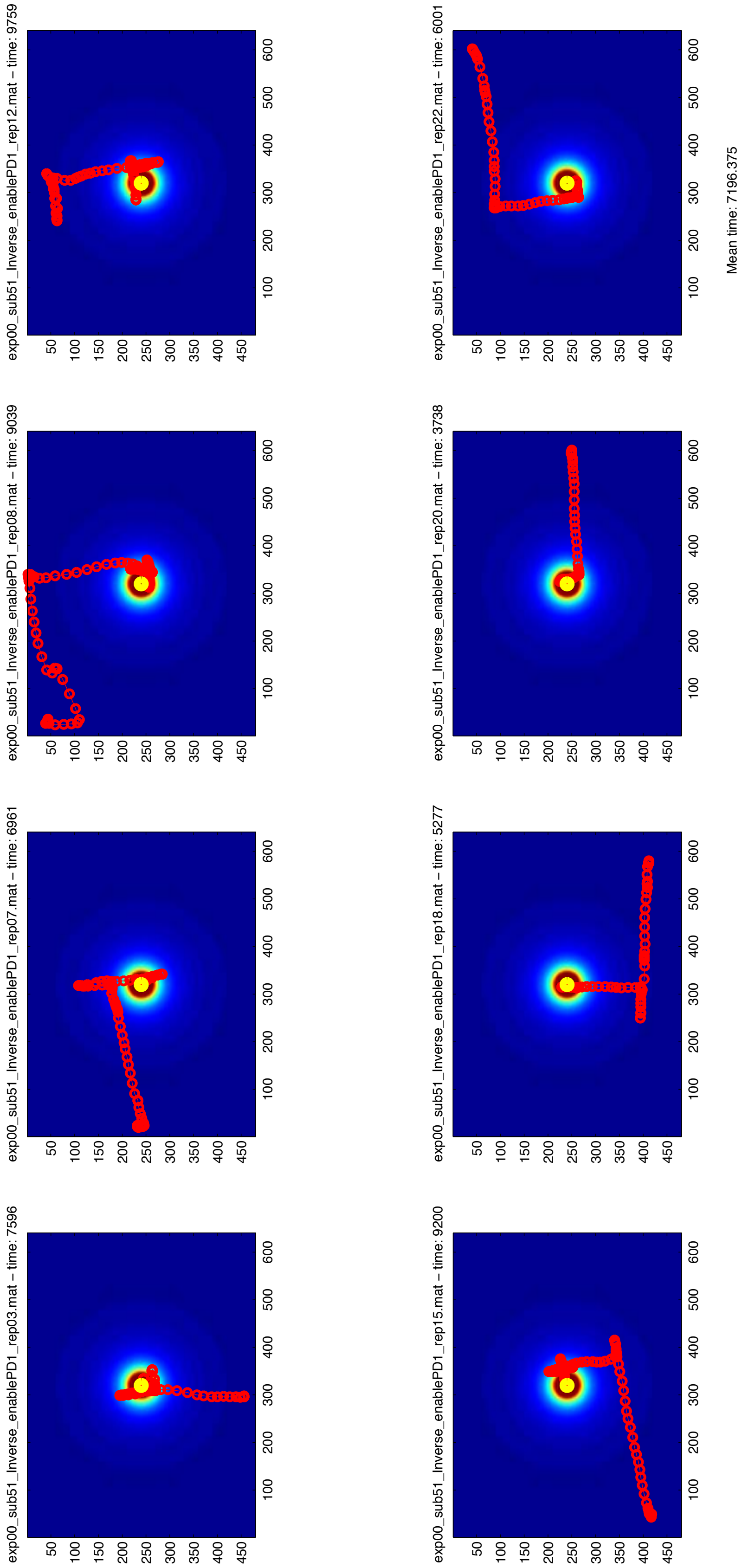
EXP 0 - ONLY TAMO



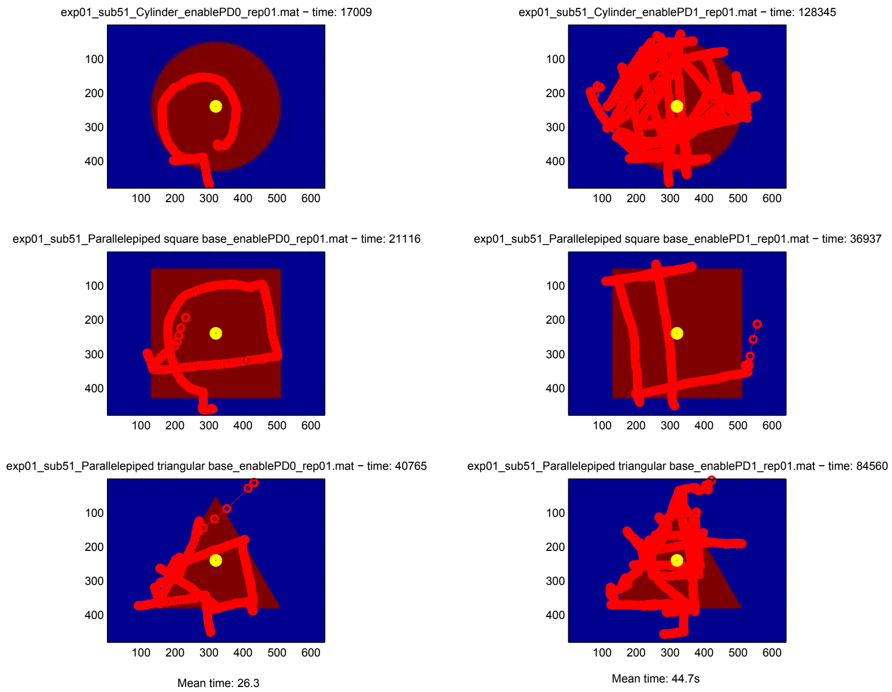
EXP 0 - ONLY AUDIO



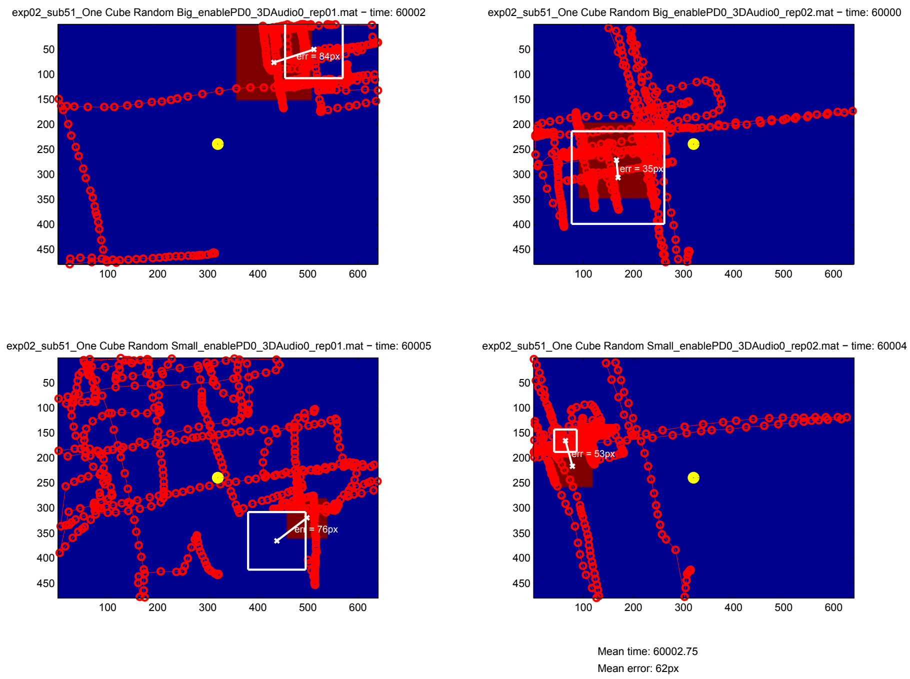
EXP 0 - TAMO AND AUDIO



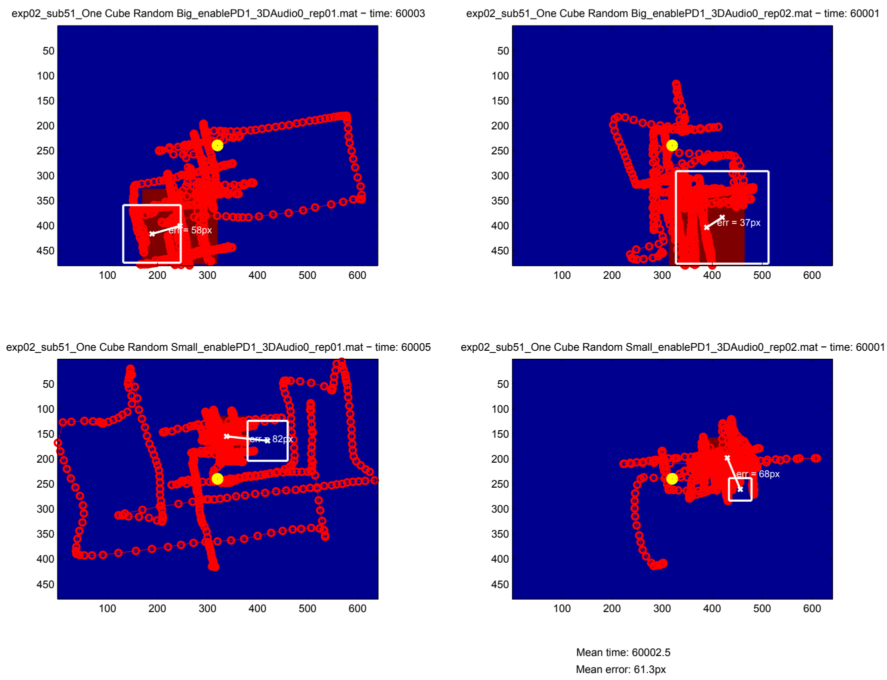
EXP 1 - NO AUDIO AGAINST AUDIO



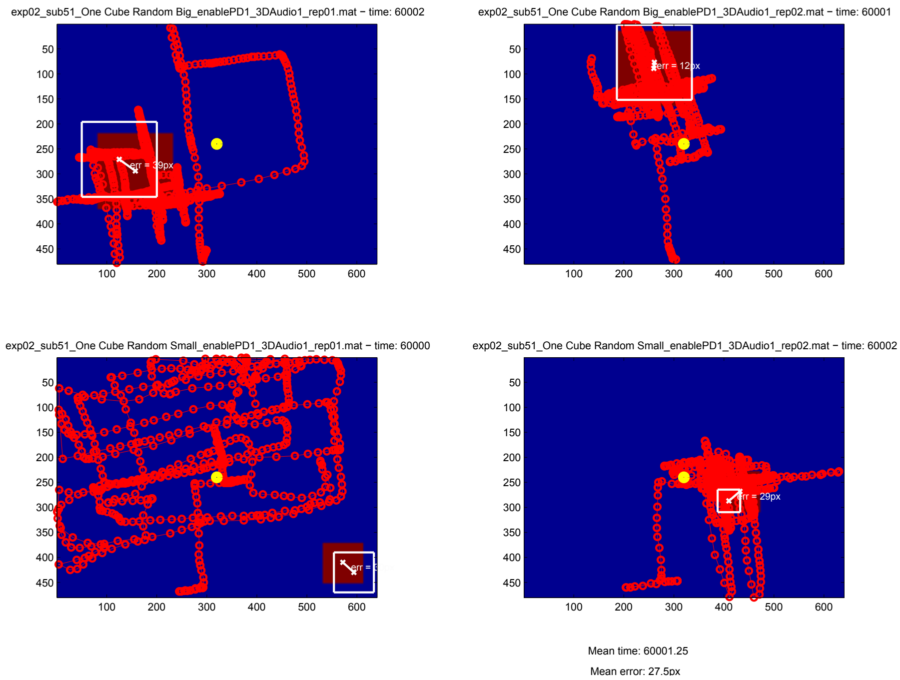
EXP 2 - ONLY TAMO



EXP 2 - TAMO AND AUDIO 2D

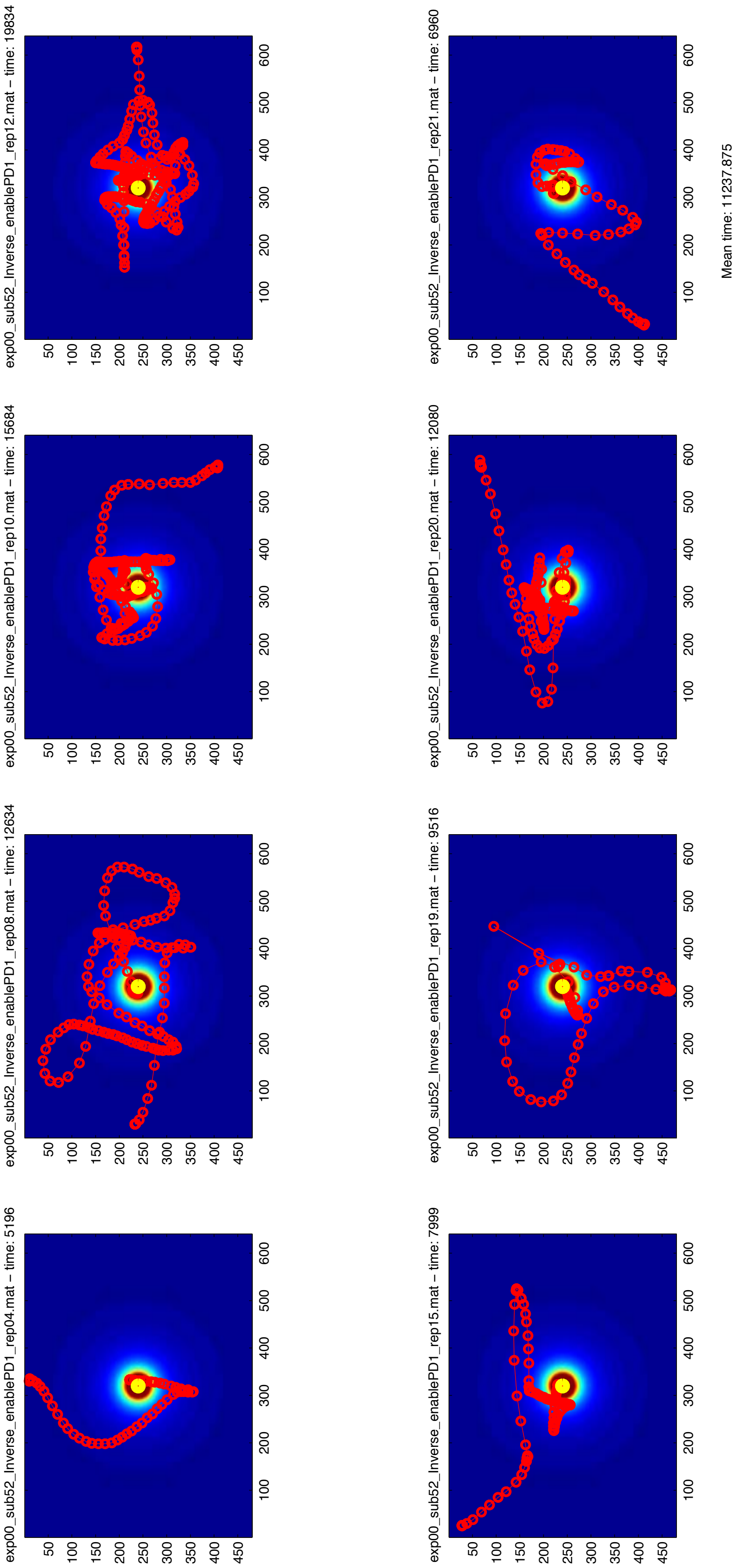


EXP 2 - TAMO AND AUDIO 3D

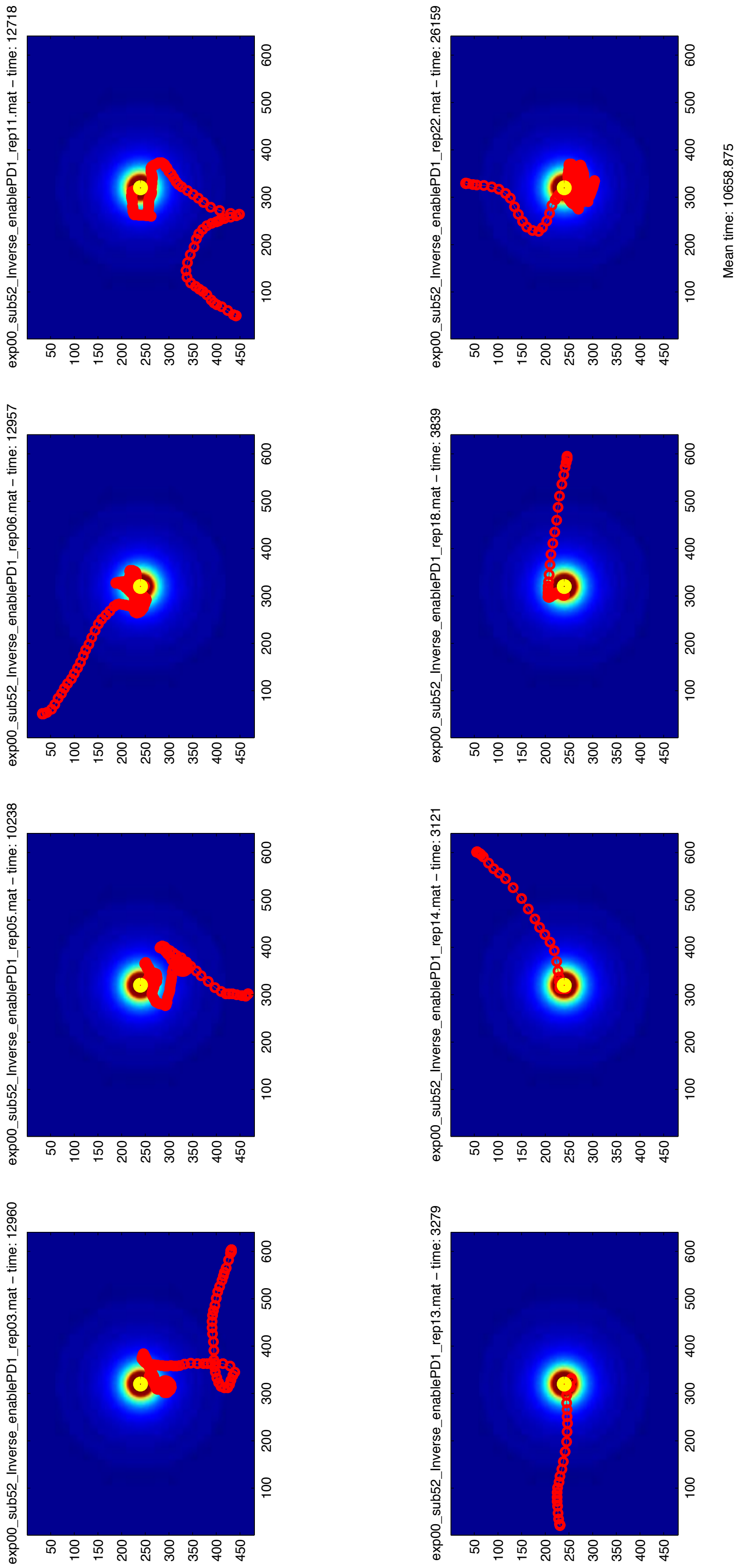


A.7 SUBJECT 52

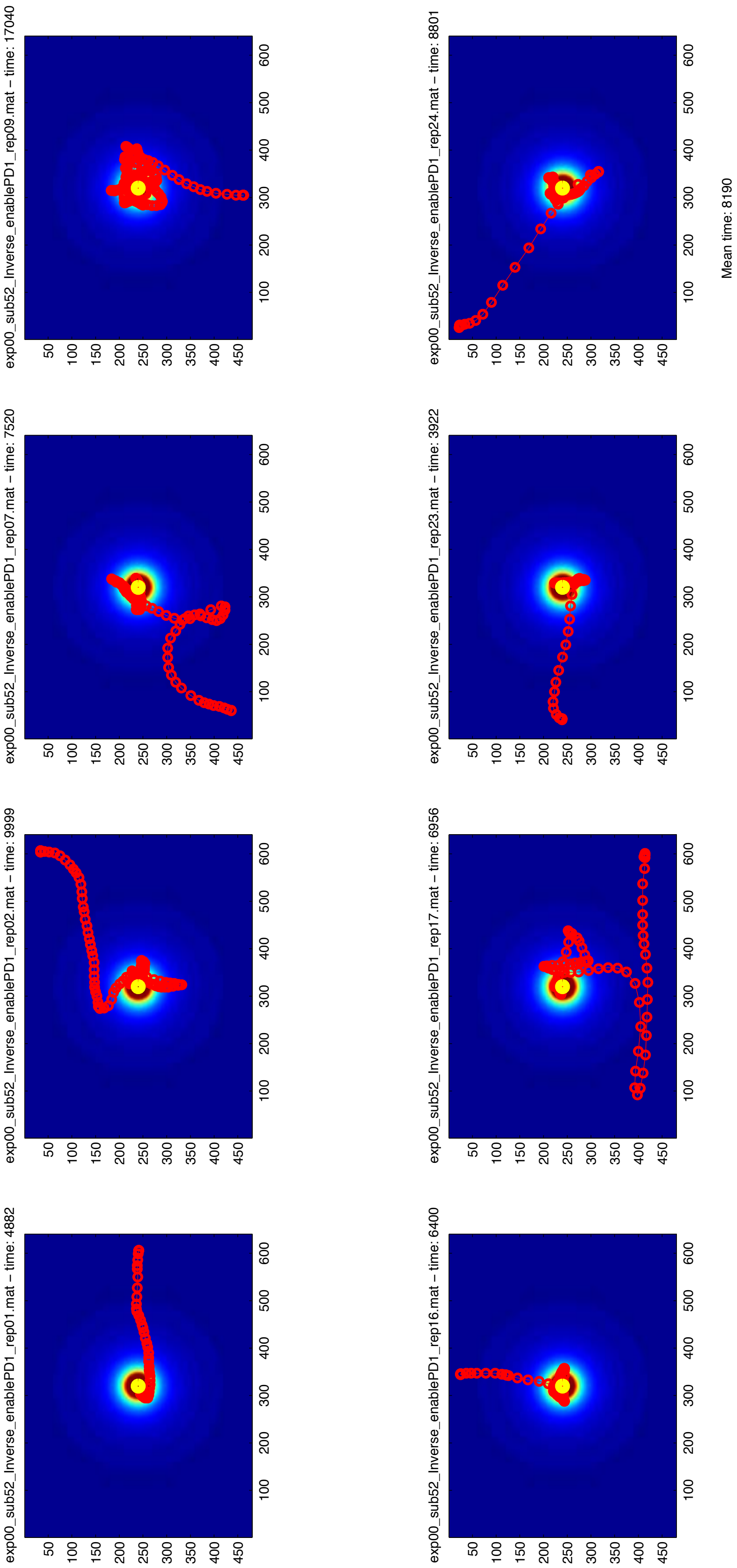
EXP 0 - ONLY TAMO



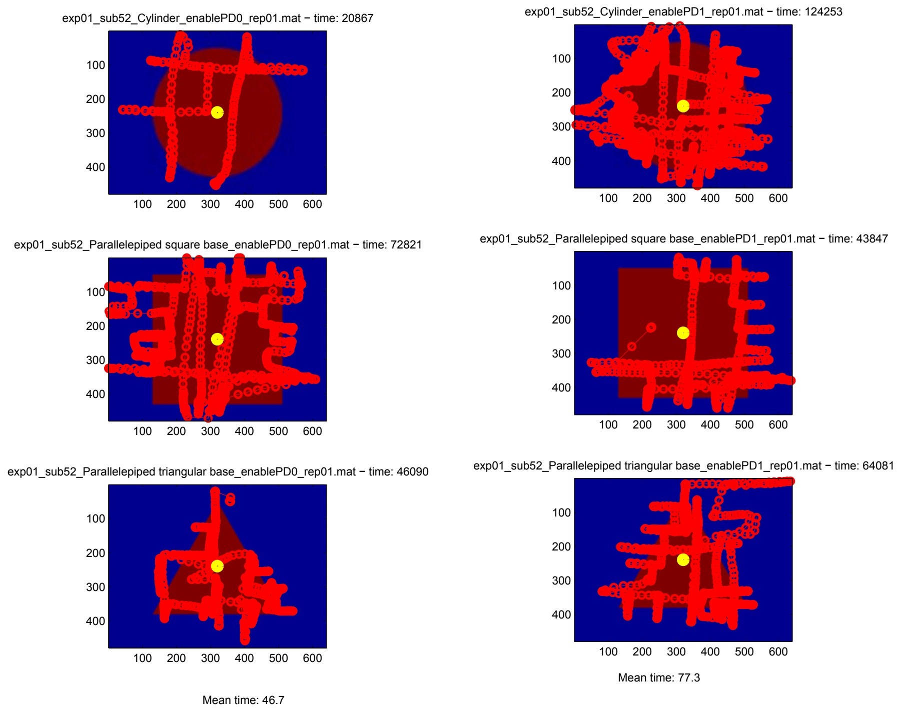
EXP 0 - ONLY AUDIO



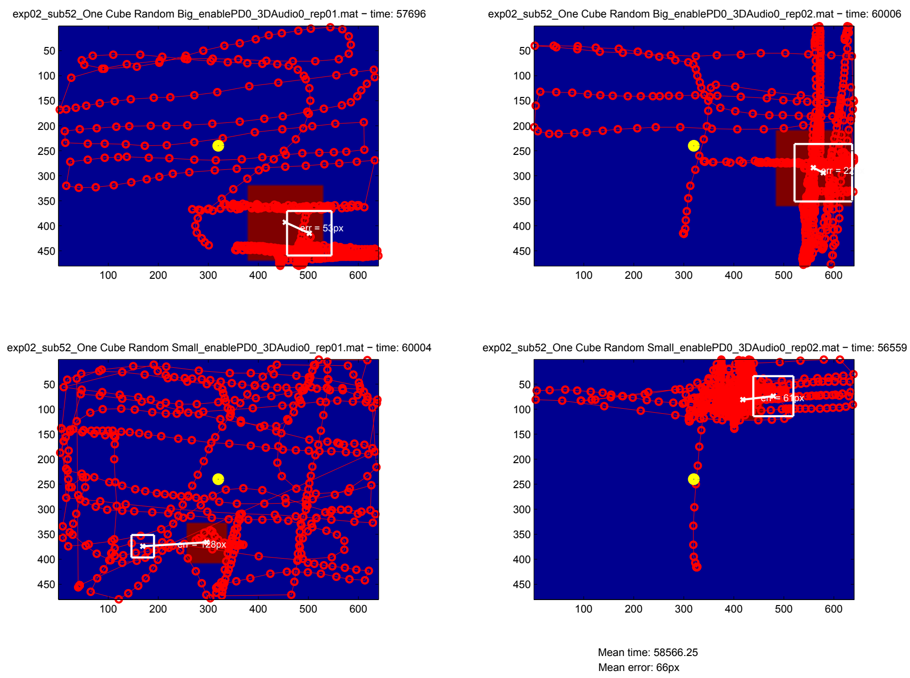
EXP 0 - TAMO AND AUDIO



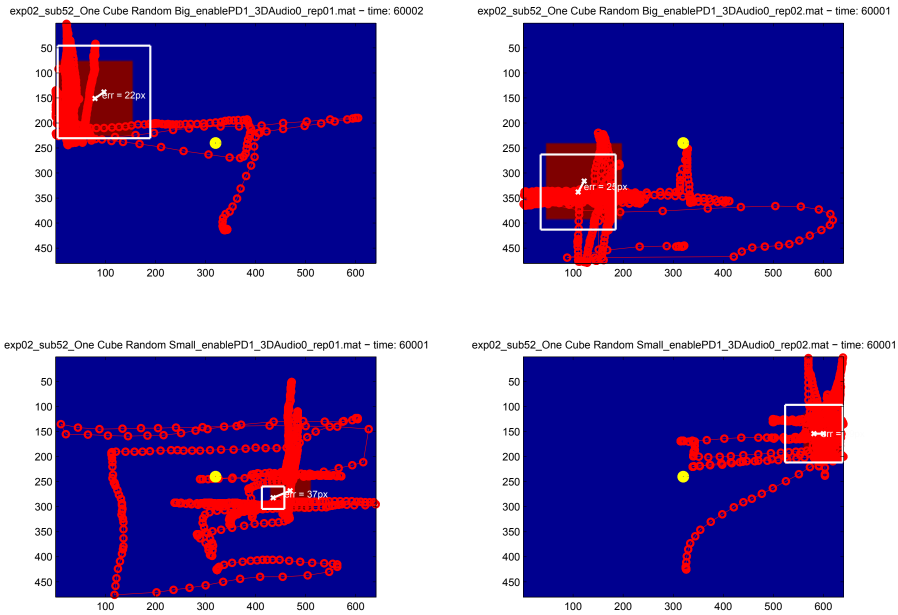
EXP 1 - NO AUDIO AGAINST AUDIO



EXP 2 - ONLY TAMO



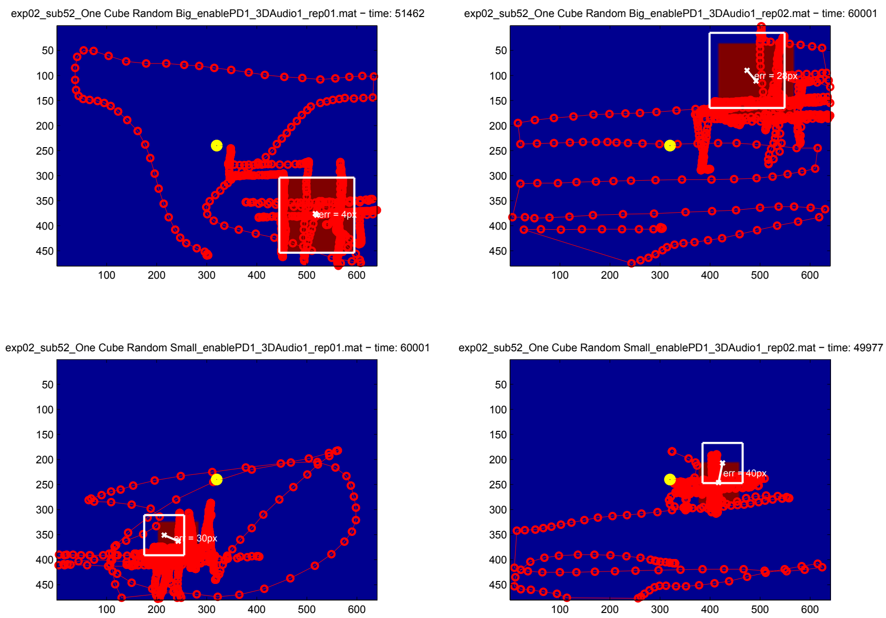
EXP 2 - TAMO AND AUDIO 2D



Mean time: 60001.25

Mean error: 25.8px

EXP 2 - TAMO AND AUDIO 3D

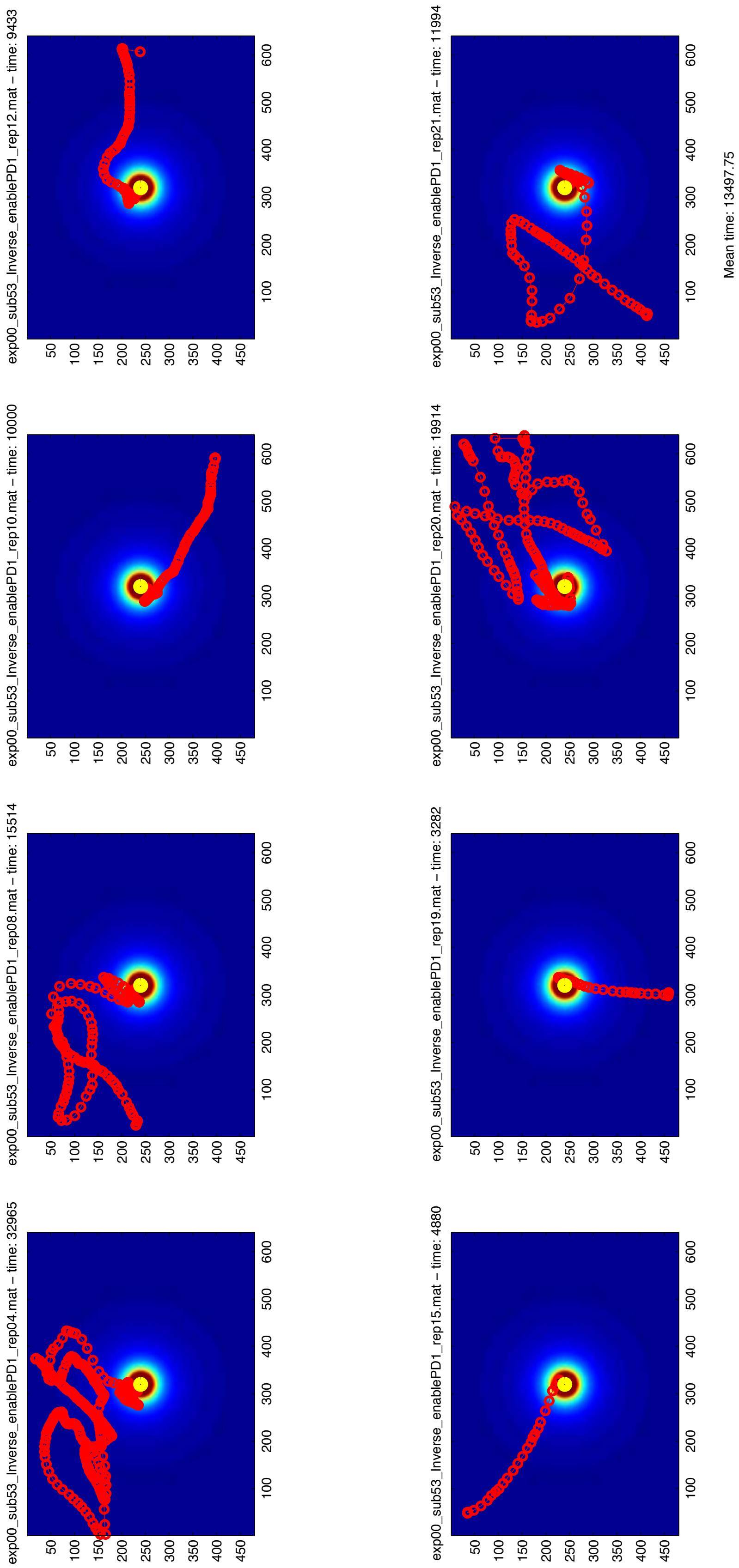


Mean time: 55360.25

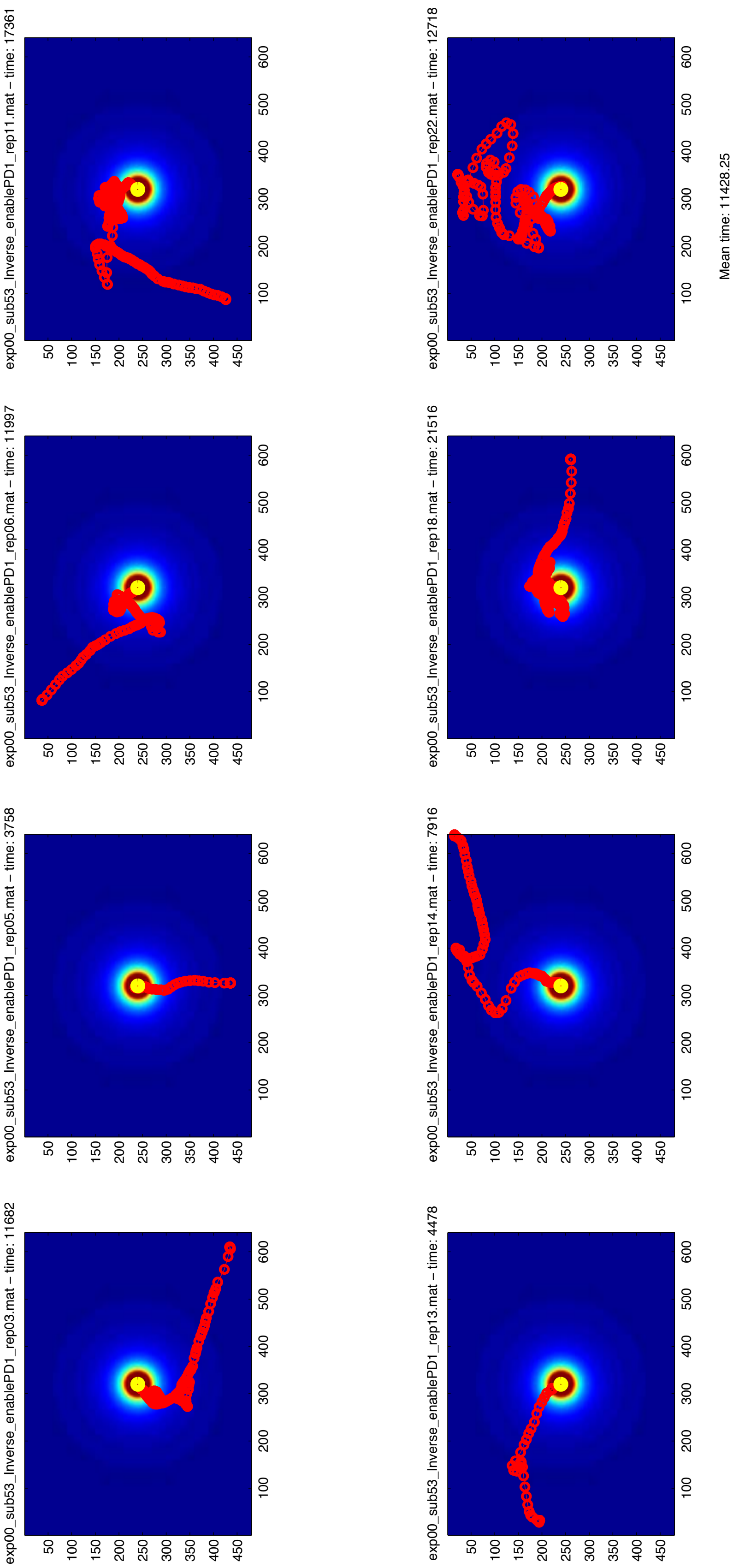
Mean error: 24.3px

A.8 SUBJECT 53

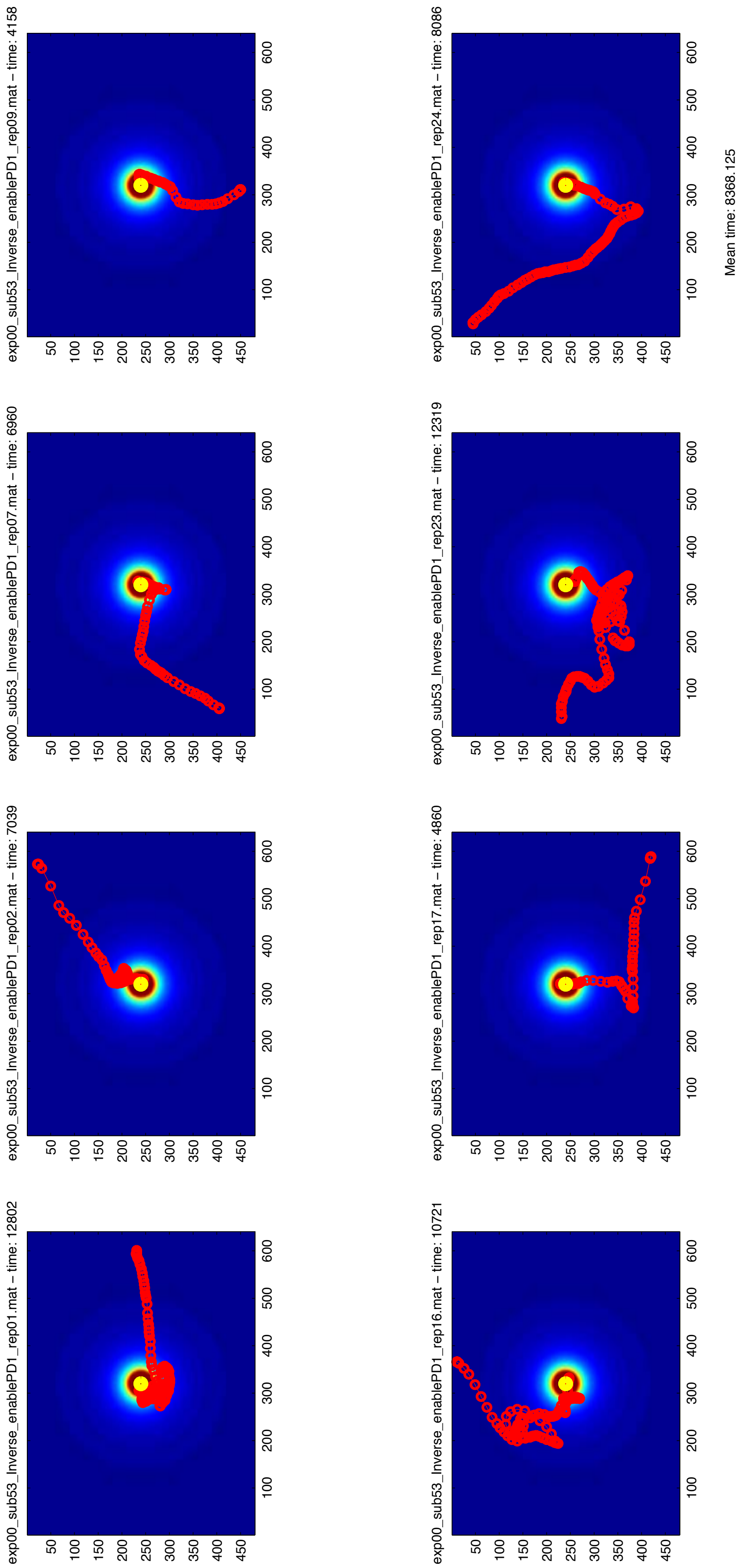
EXP 0 - ONLY TAMO



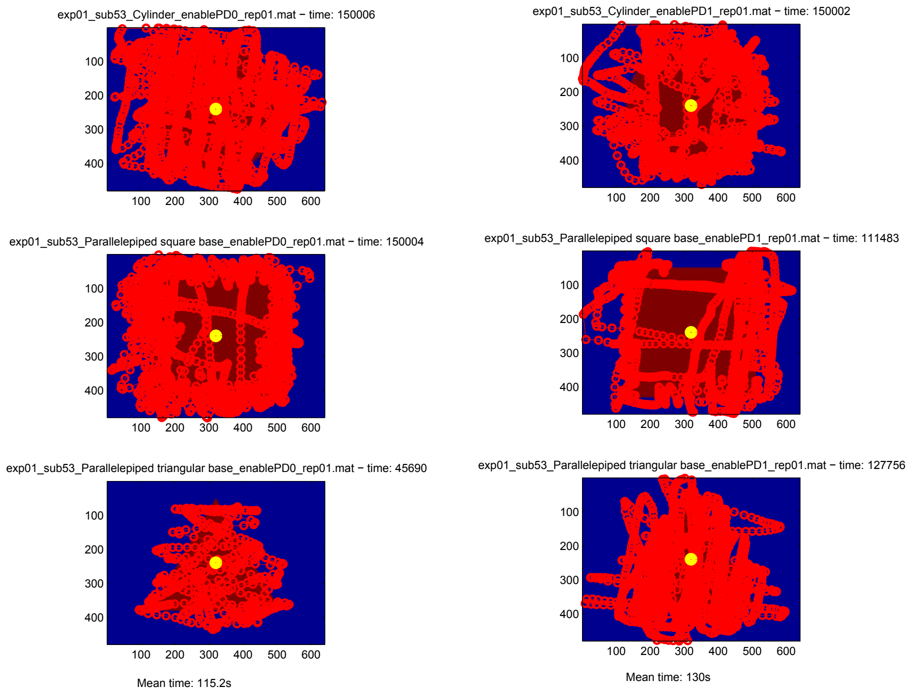
EXP 0 - ONLY AUDIO



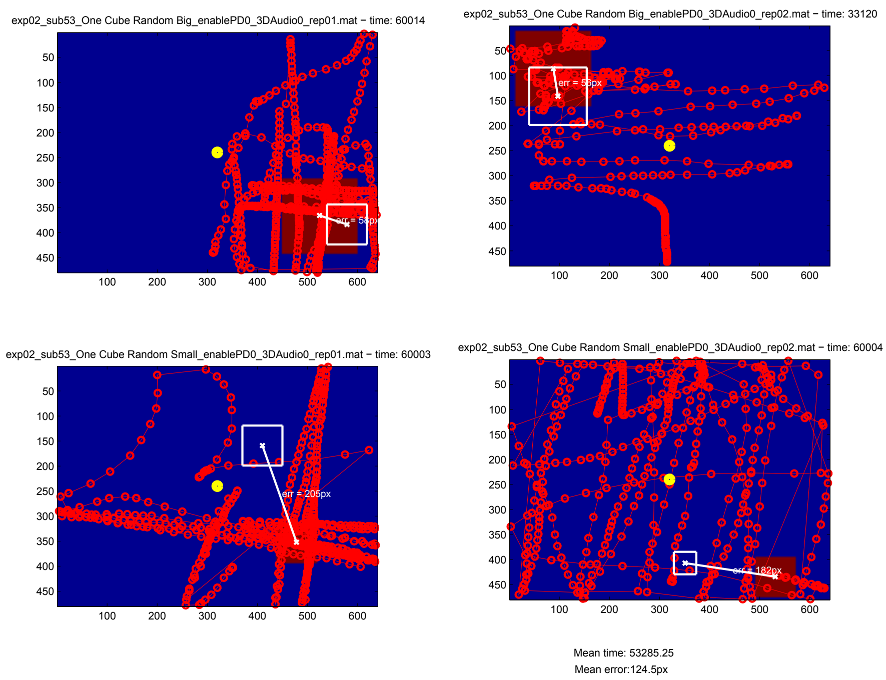
EXP 0 - TAMO AND AUDIO



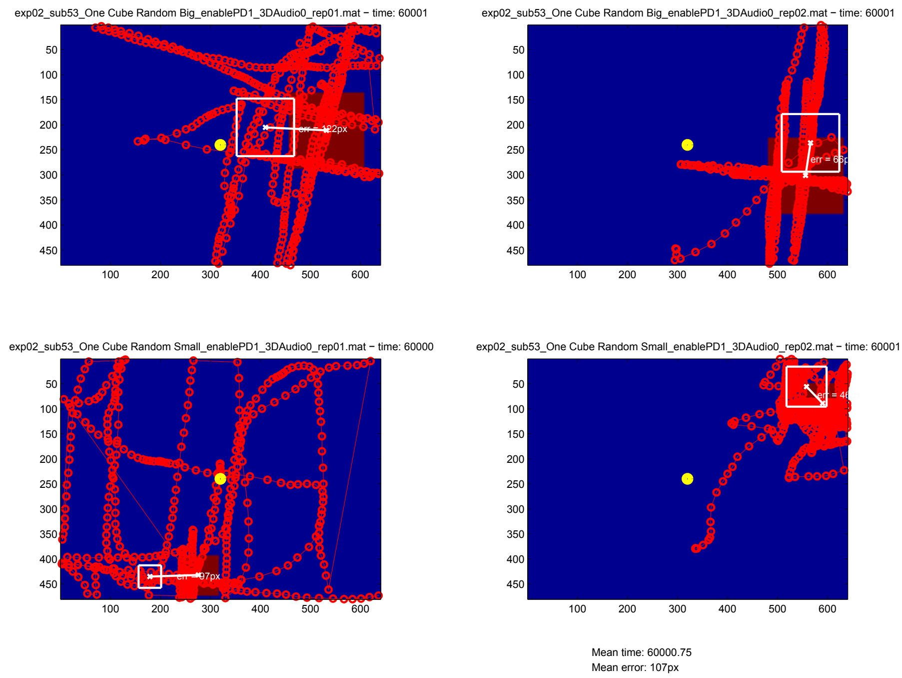
EXP 1 - NO AUDIO AGAINST AUDIO



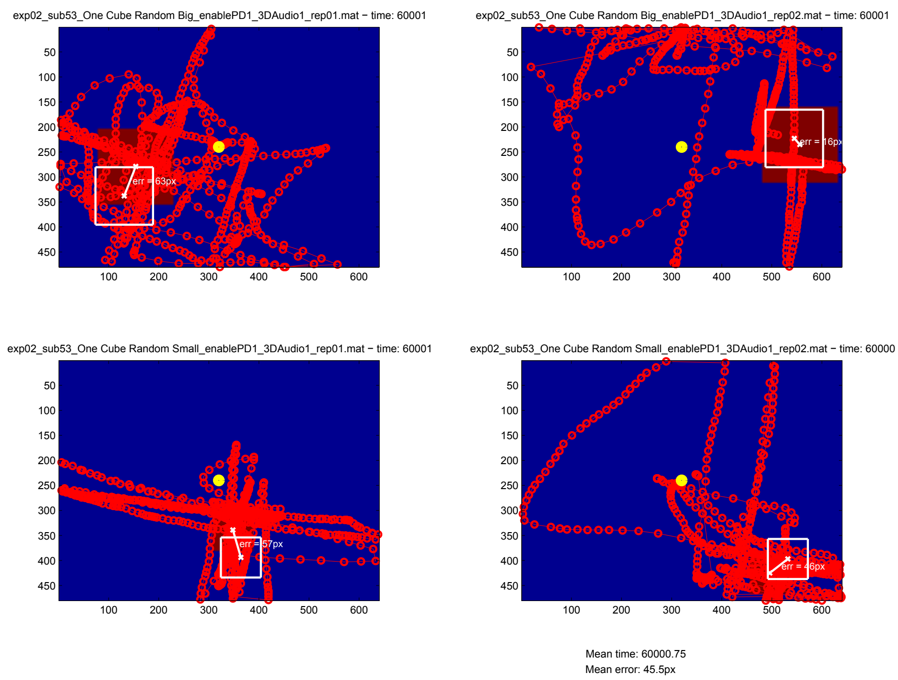
EXP 2 - ONLY TAMO



EXP 2 - TAMO AND AUDIO 2D

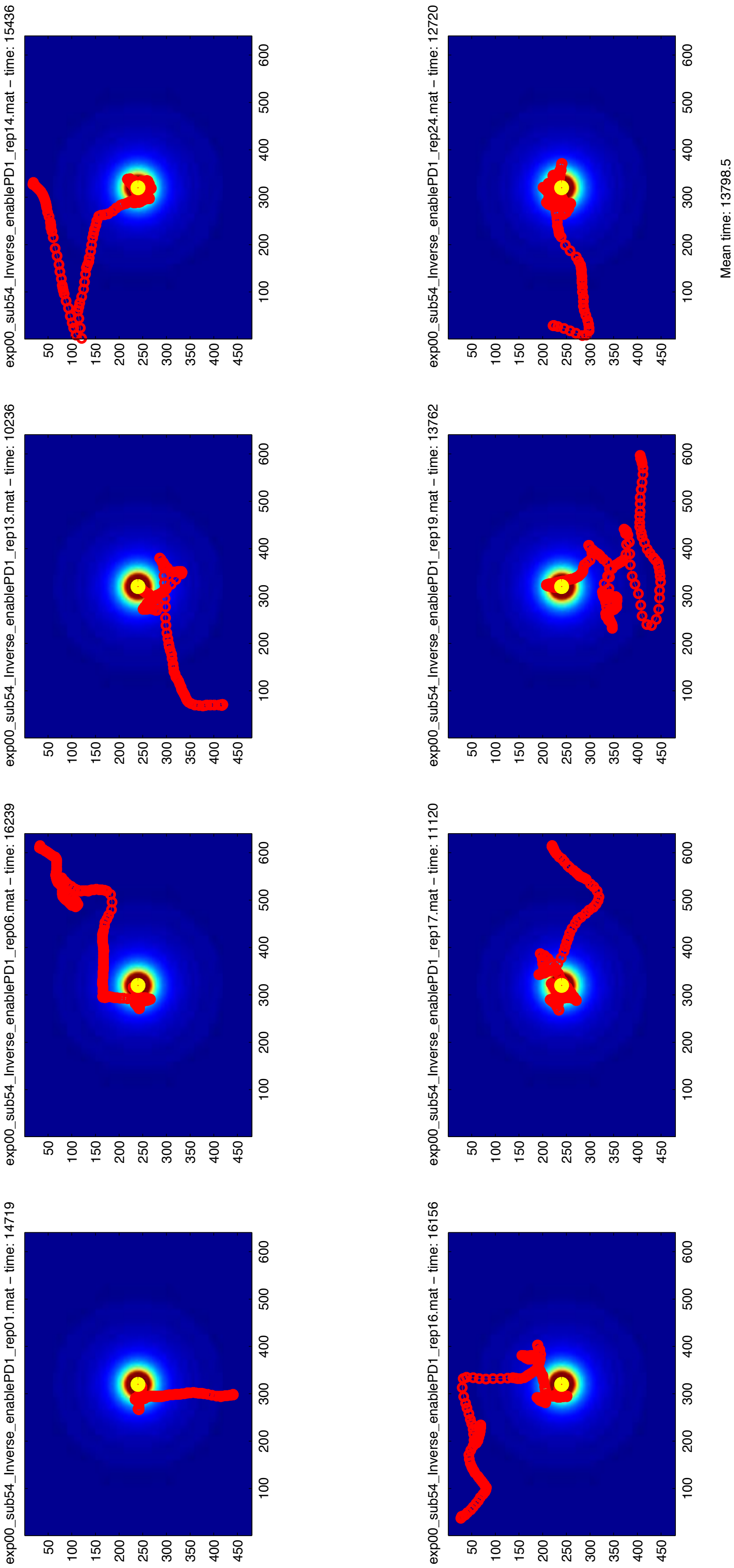


EXP 2 - TAMO AND AUDIO 3D

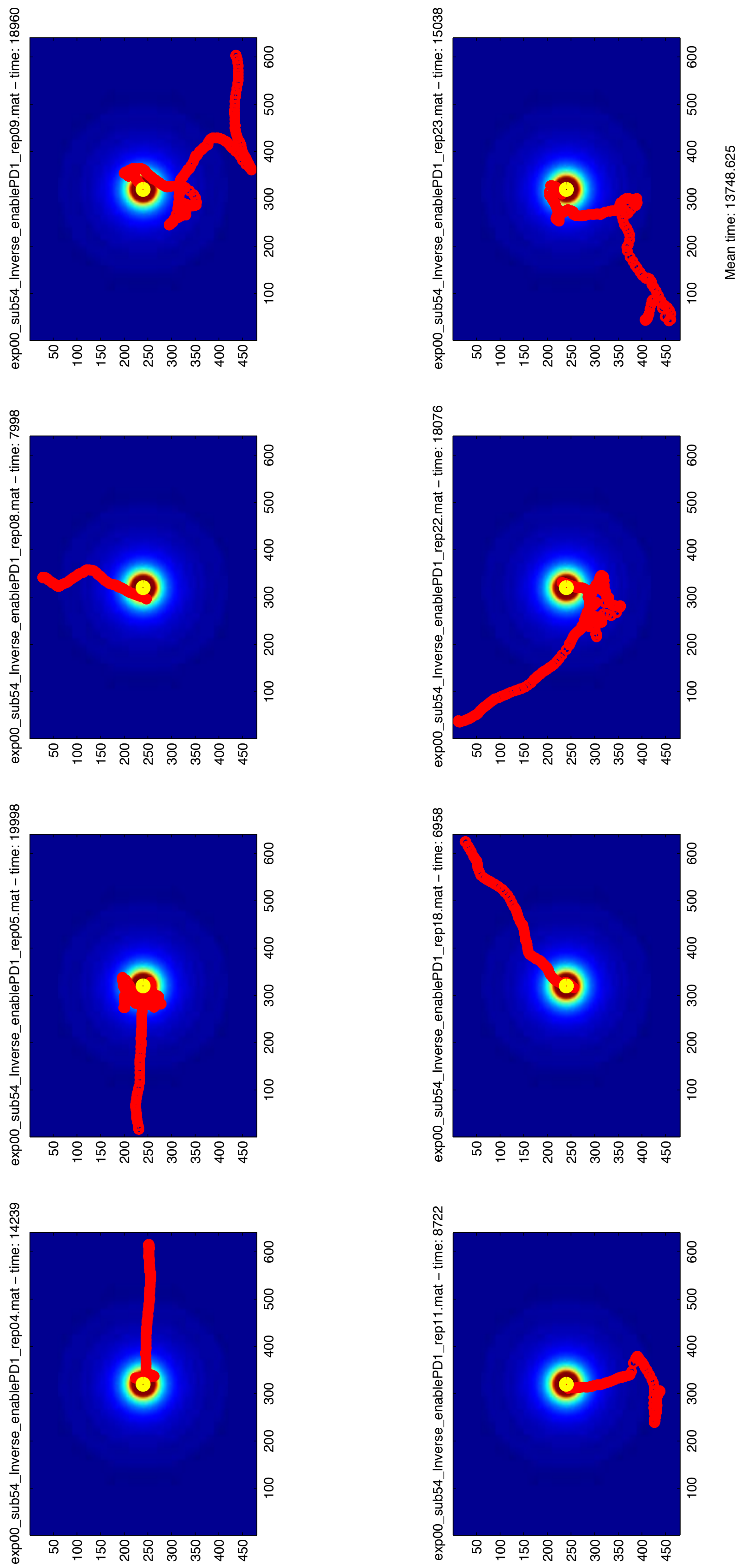


A.9 SUBJECT 54

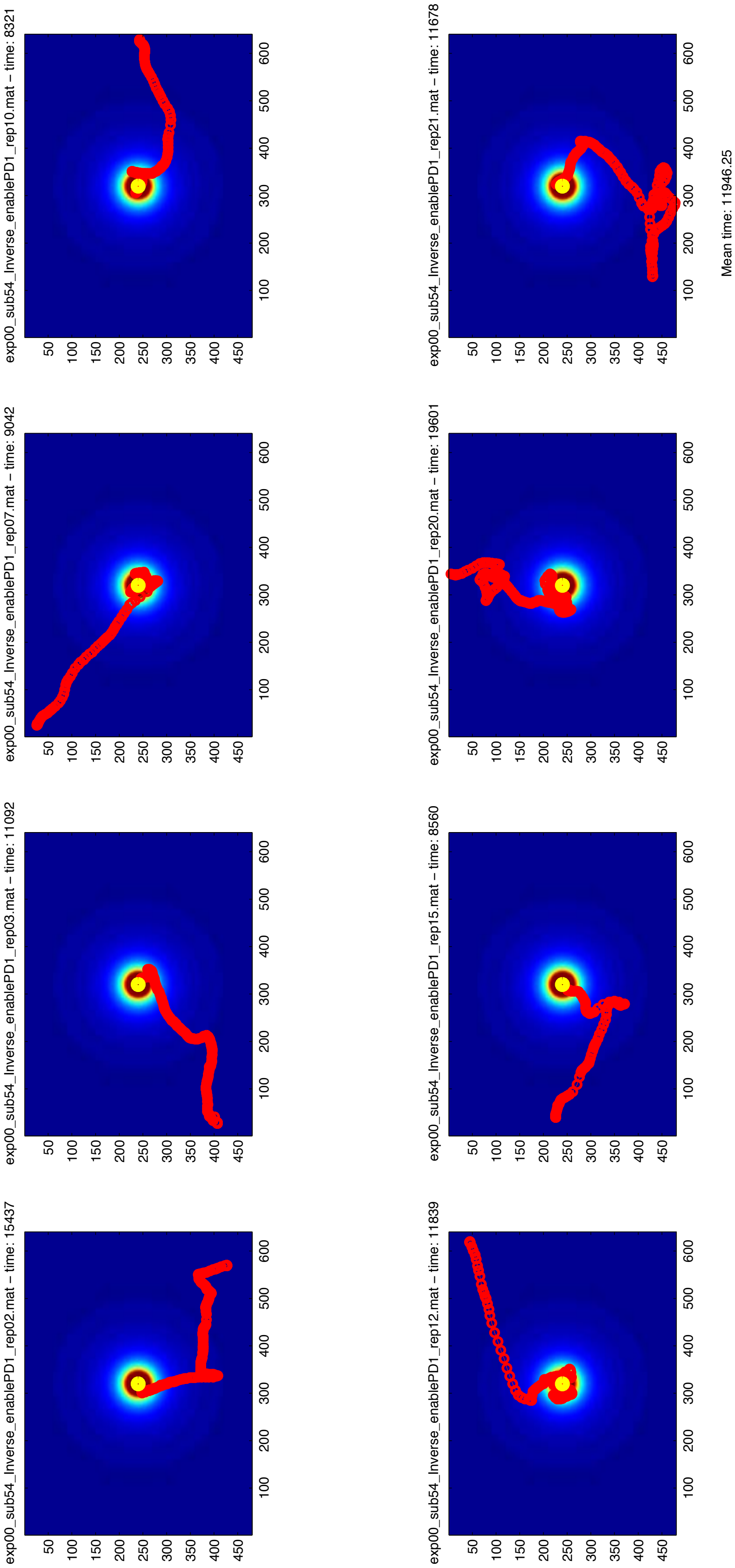
EXP 0 - ONLY TAMO



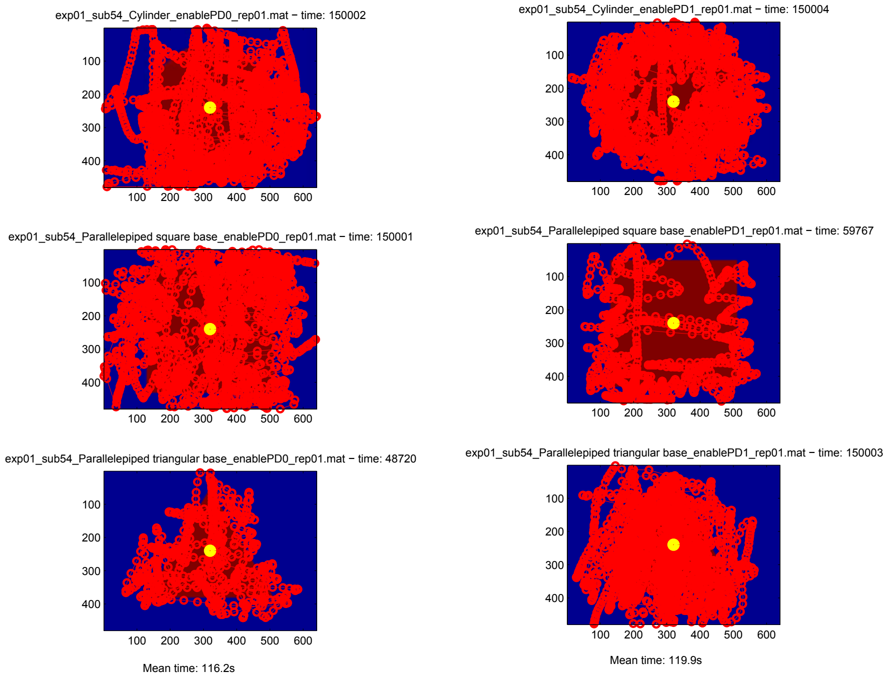
EXP 0 - ONLY AUDIO



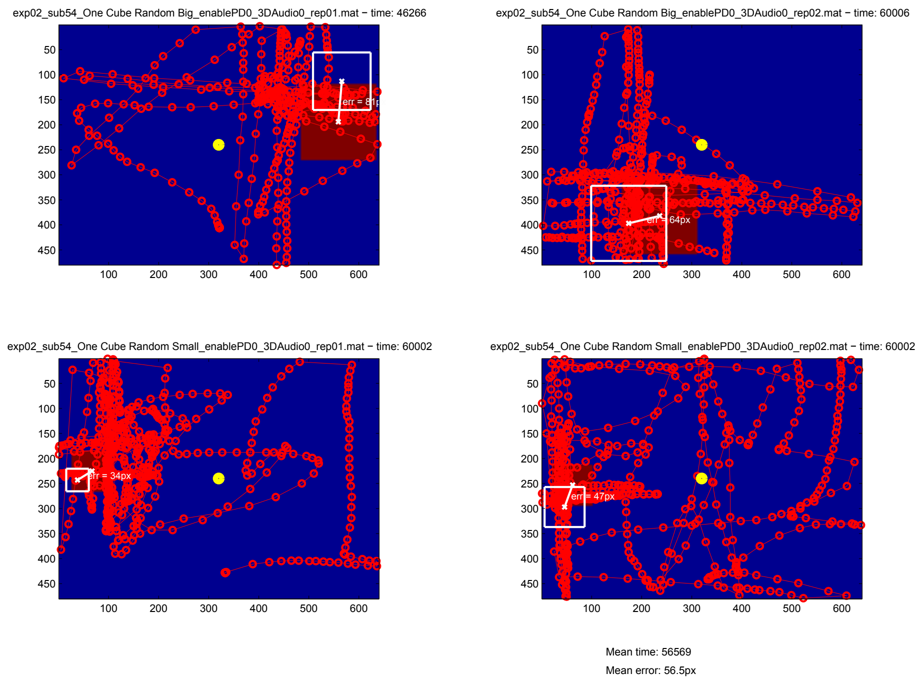
EXP 0 - TAMO AND AUDIO



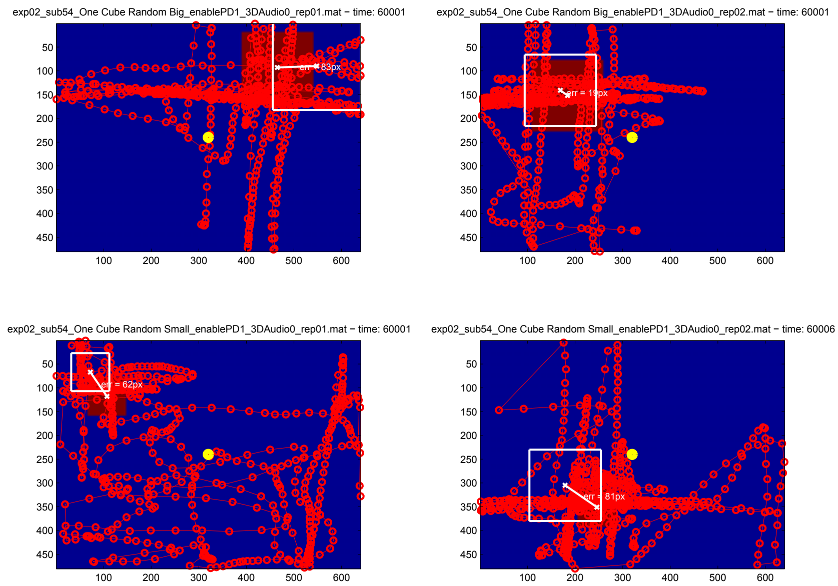
EXP 1 - NO AUDIO AGAINST AUDIO



EXP 2 - ONLY TAMO

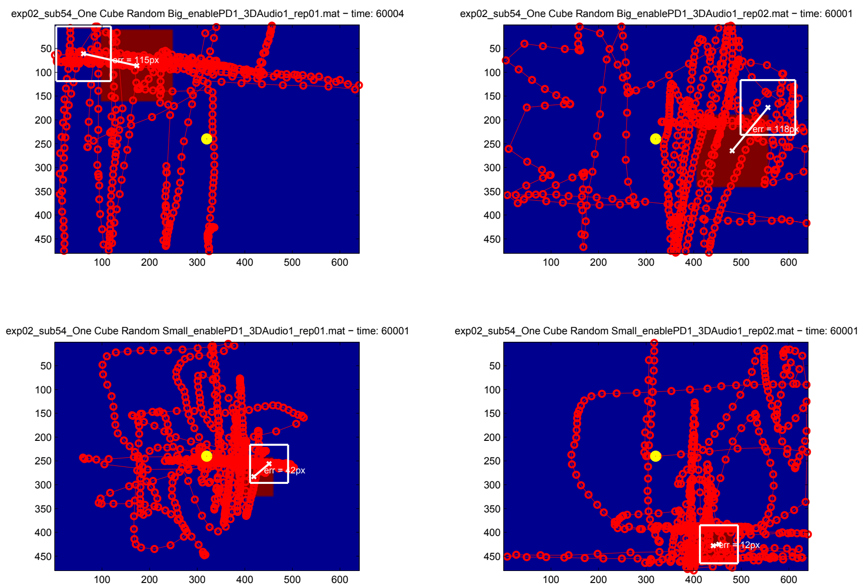


EXP 2 - TAMO AND AUDIO 2D



Mean time: 60002.25
Mean error: 61.3px

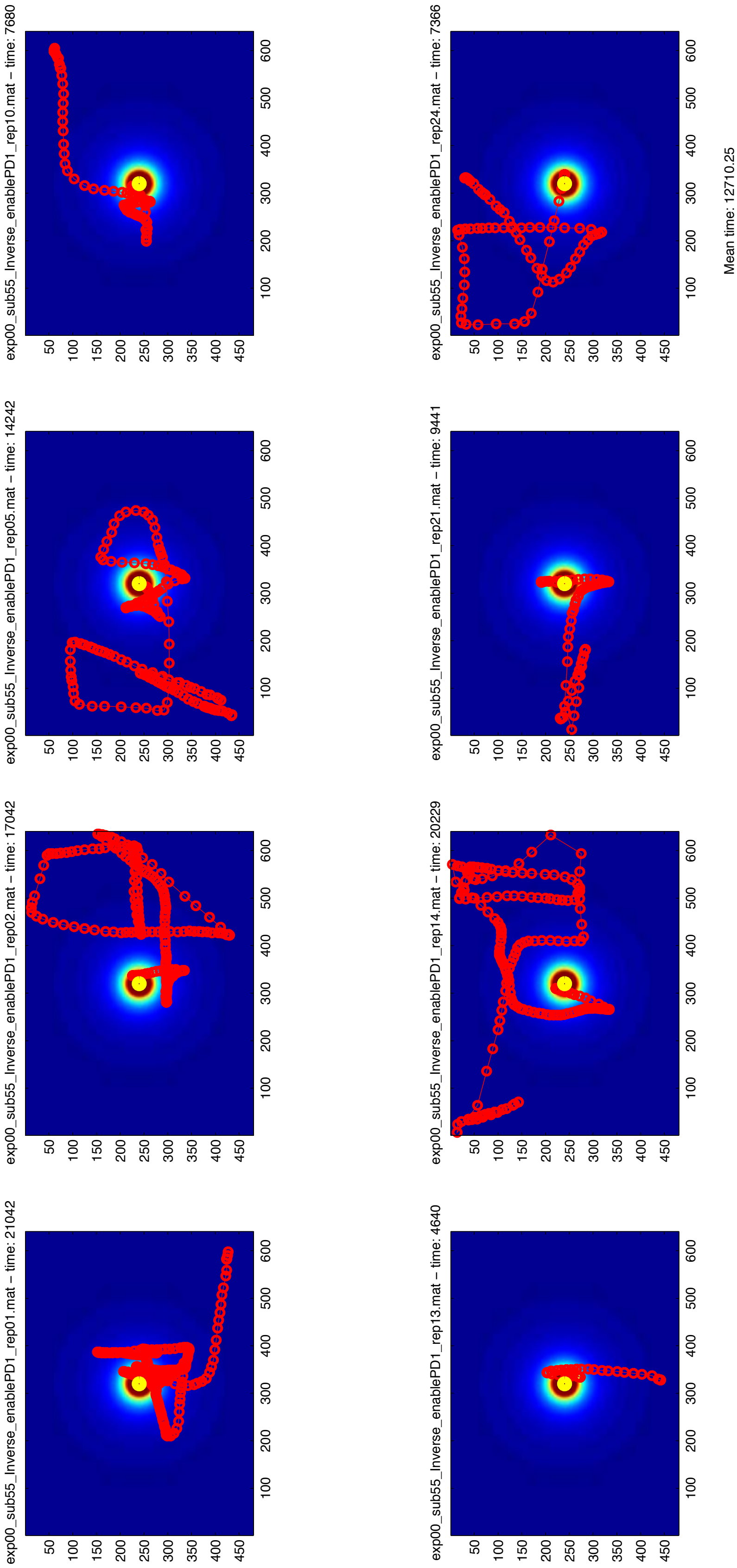
EXP 2 - TAMO AND AUDIO 3D



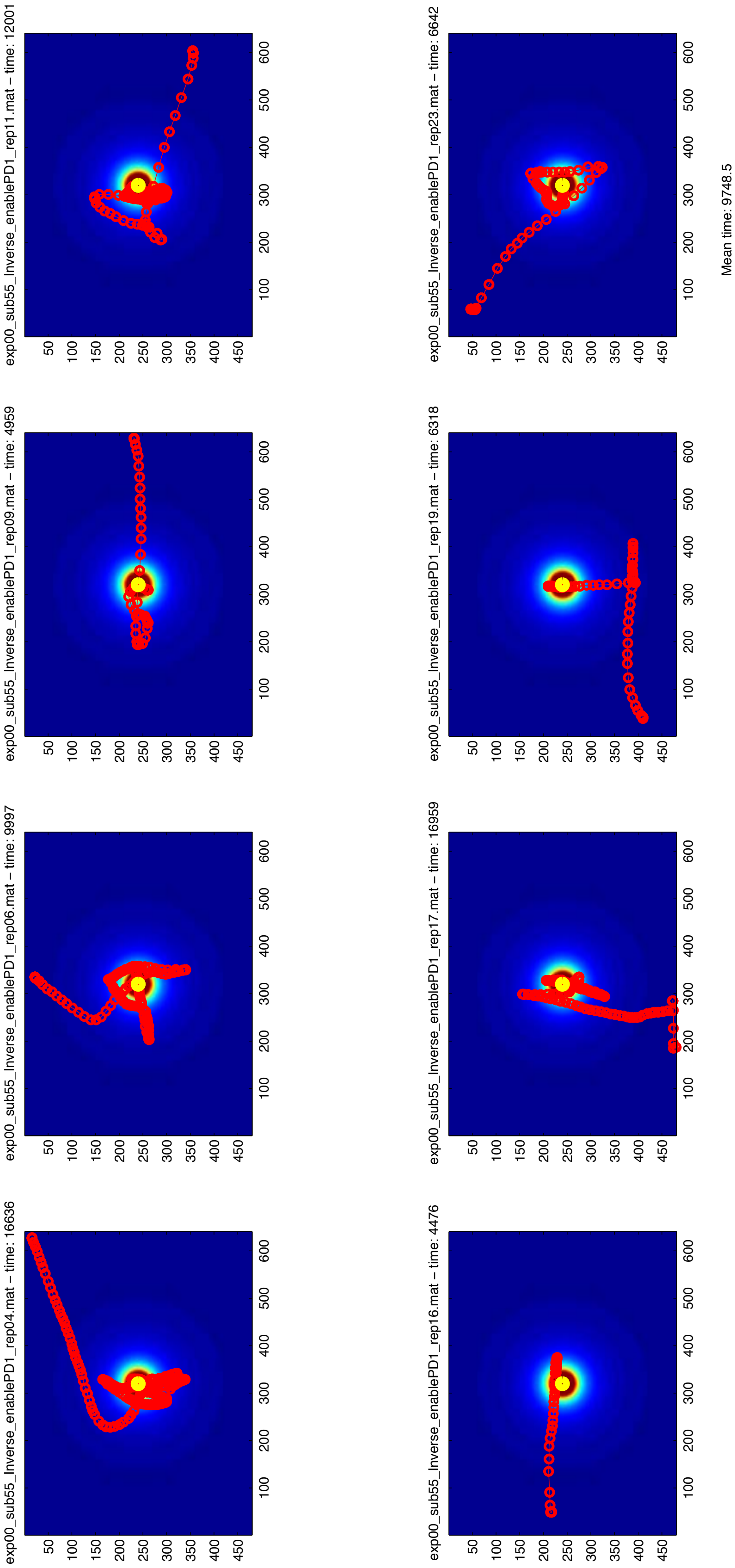
Mean time: 60001.75
Mean error: 71.8px

A.10 SUBJECT 55

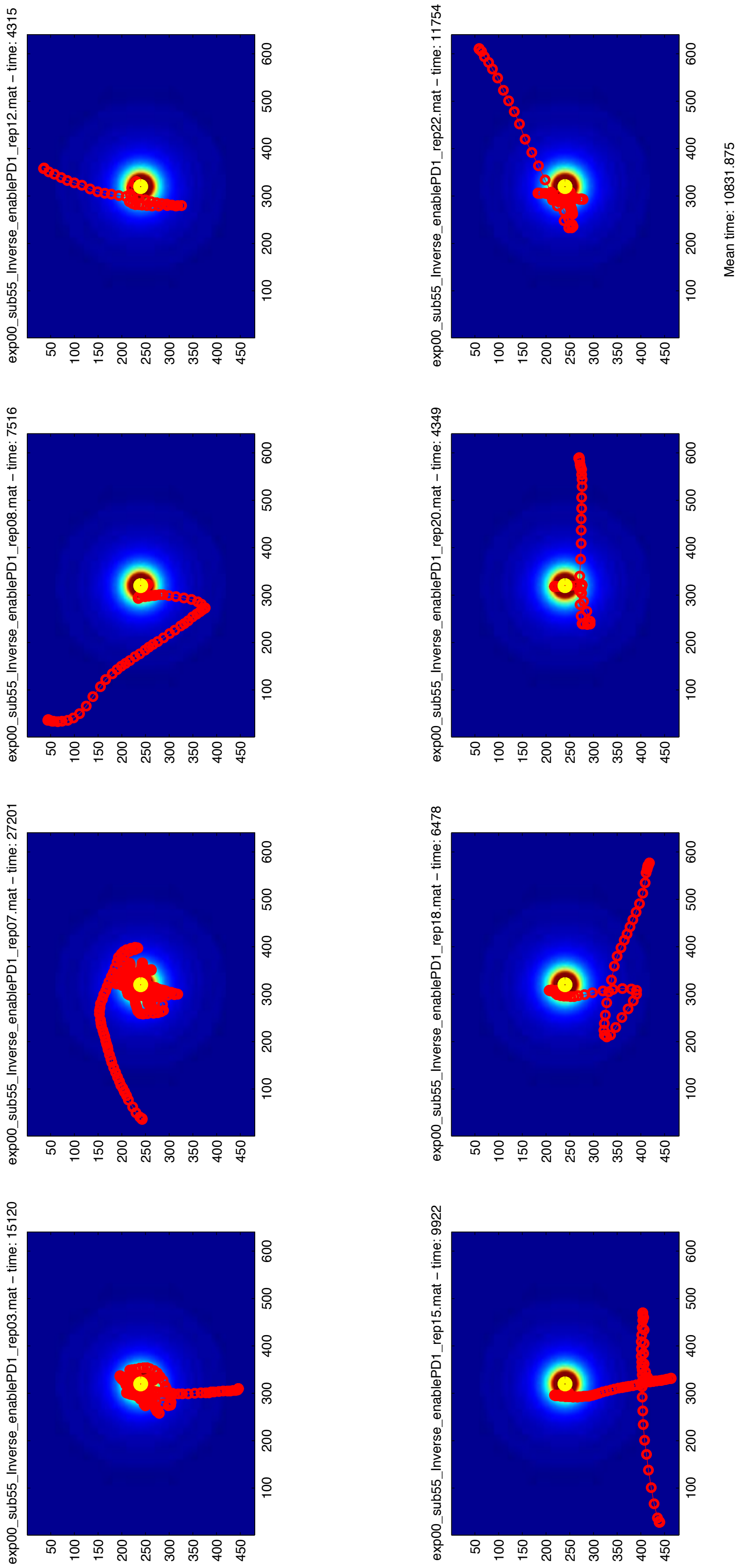
EXP 0 - ONLY TAMO



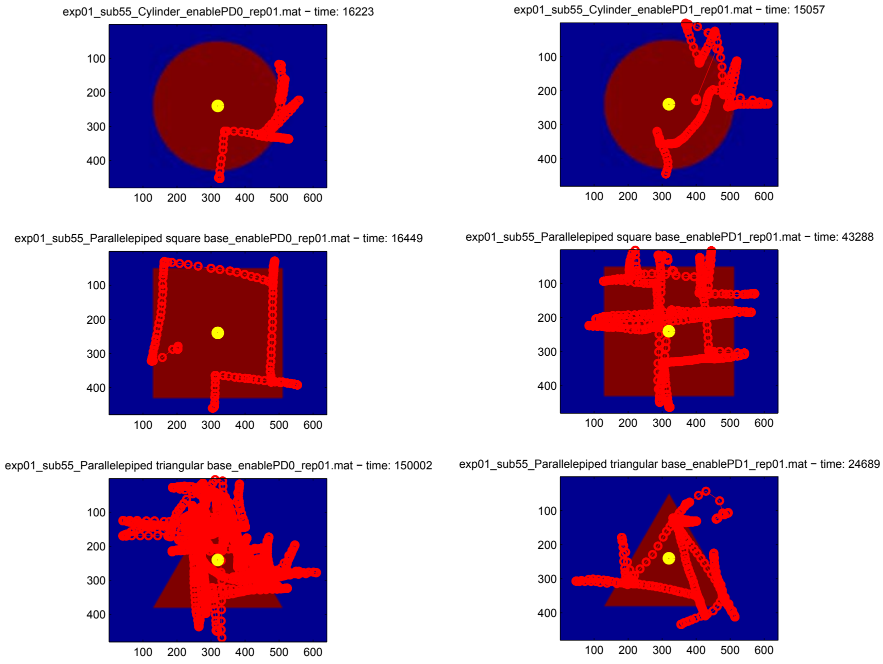
exp 0 - only audio



EXP 0 - TAMO AND AUDIO



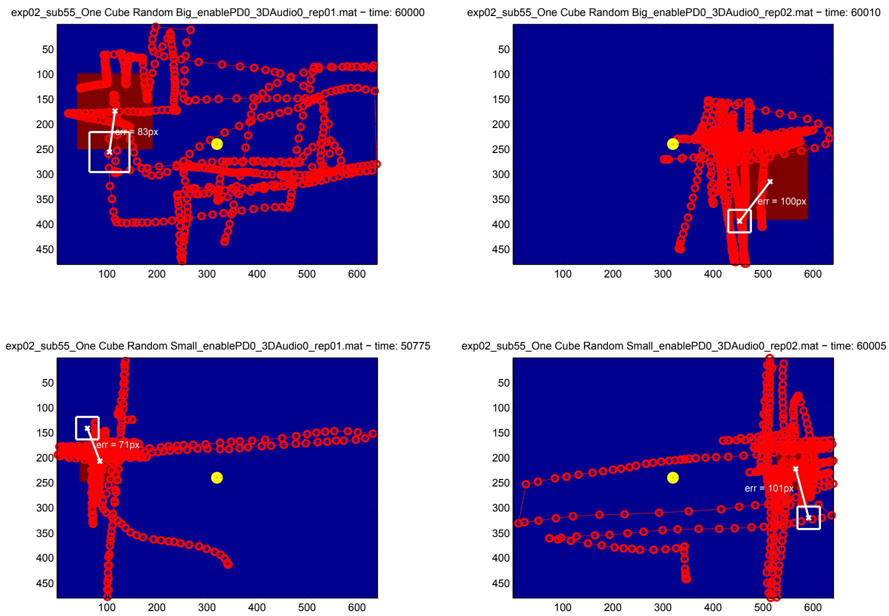
EXP 1 - NO AUDIO AGAINST AUDIO



Mean time: 60.8s

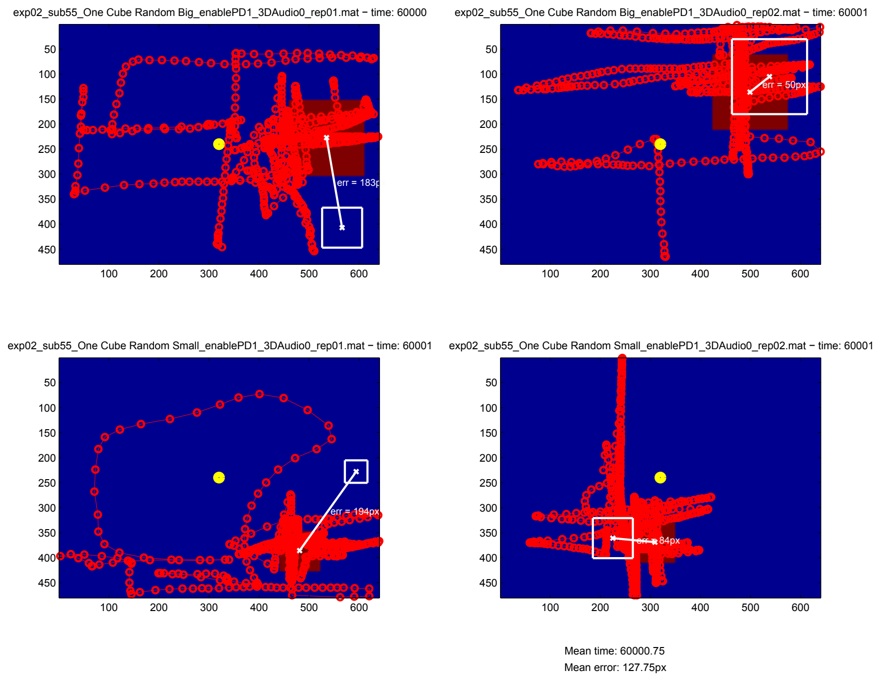
Mean time: 27.6

EXP 2 - ONLY TAMO

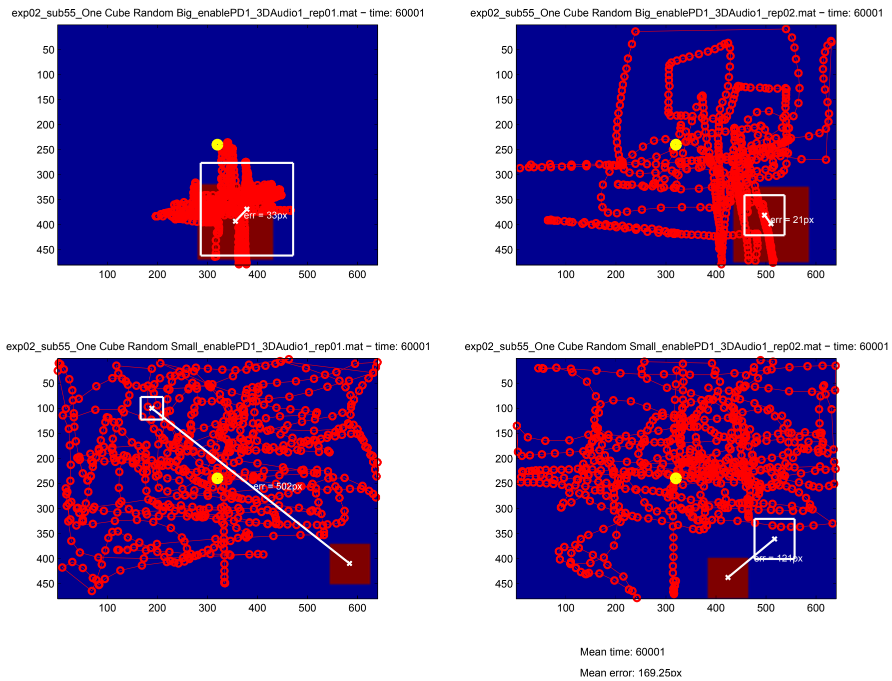


Mean time: 57697.5
Mean error: 88.7

EXP 2 - TAMO AND AUDIO 2D

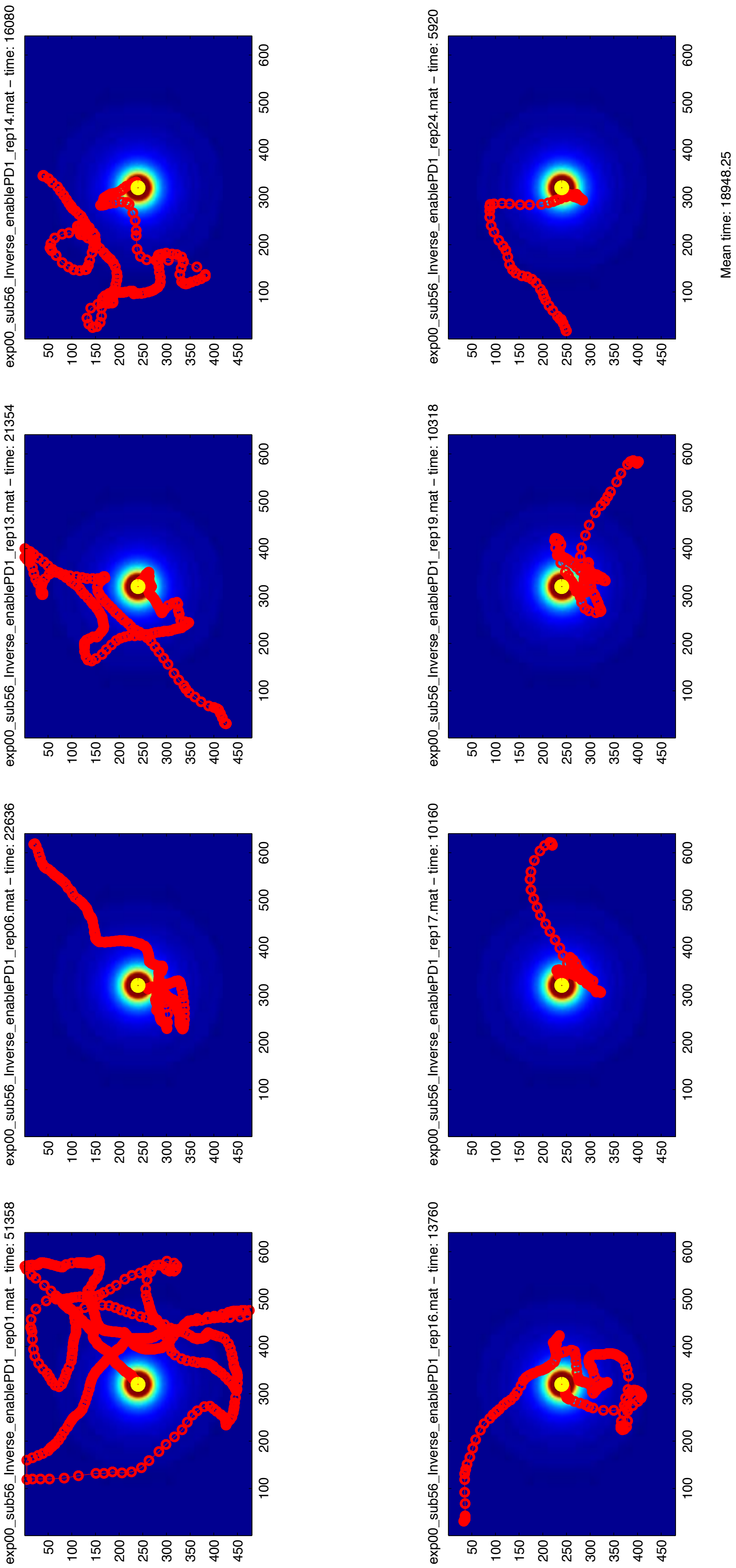


EXP 2 - TAMO AND AUDIO 3D

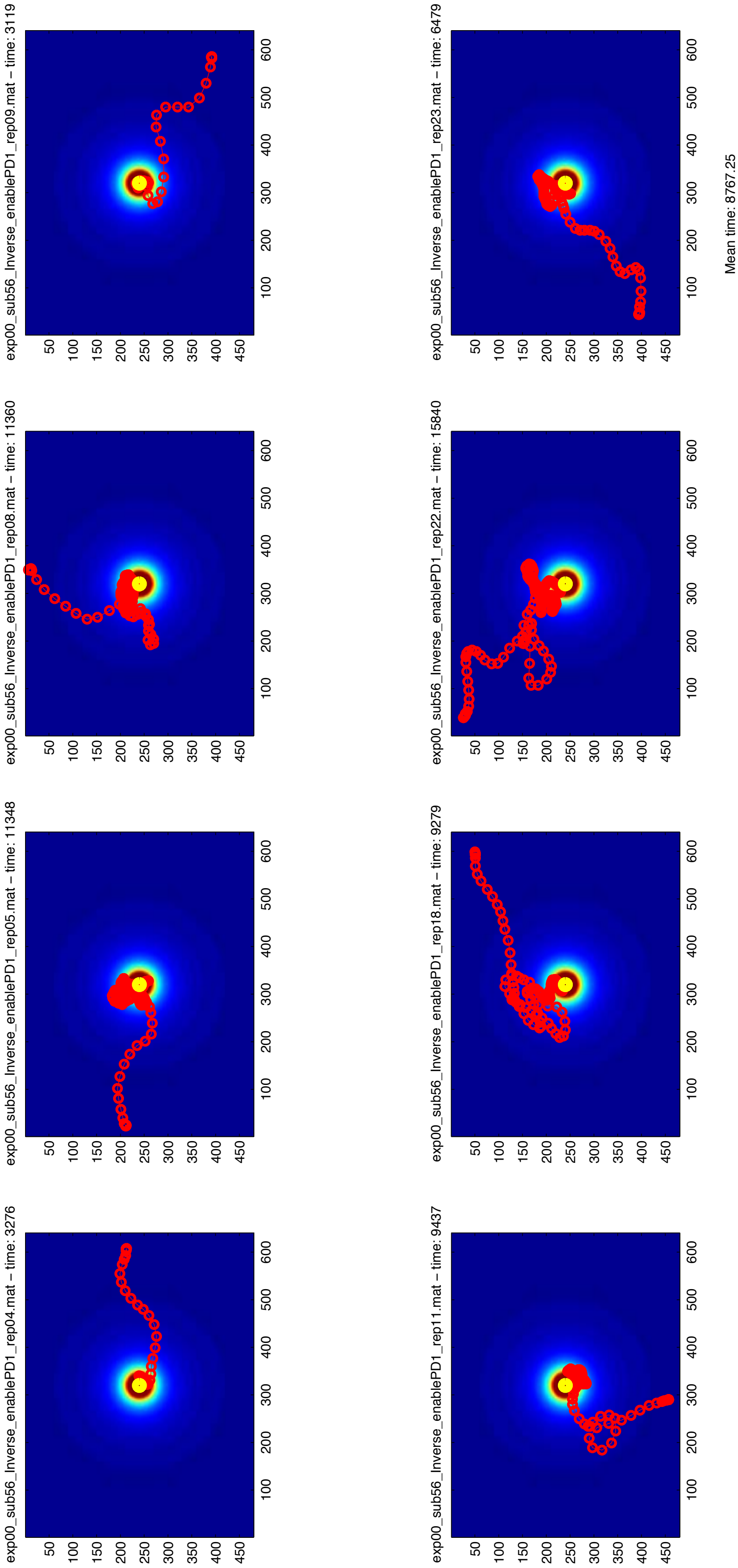


A.11 SUBJECT 56

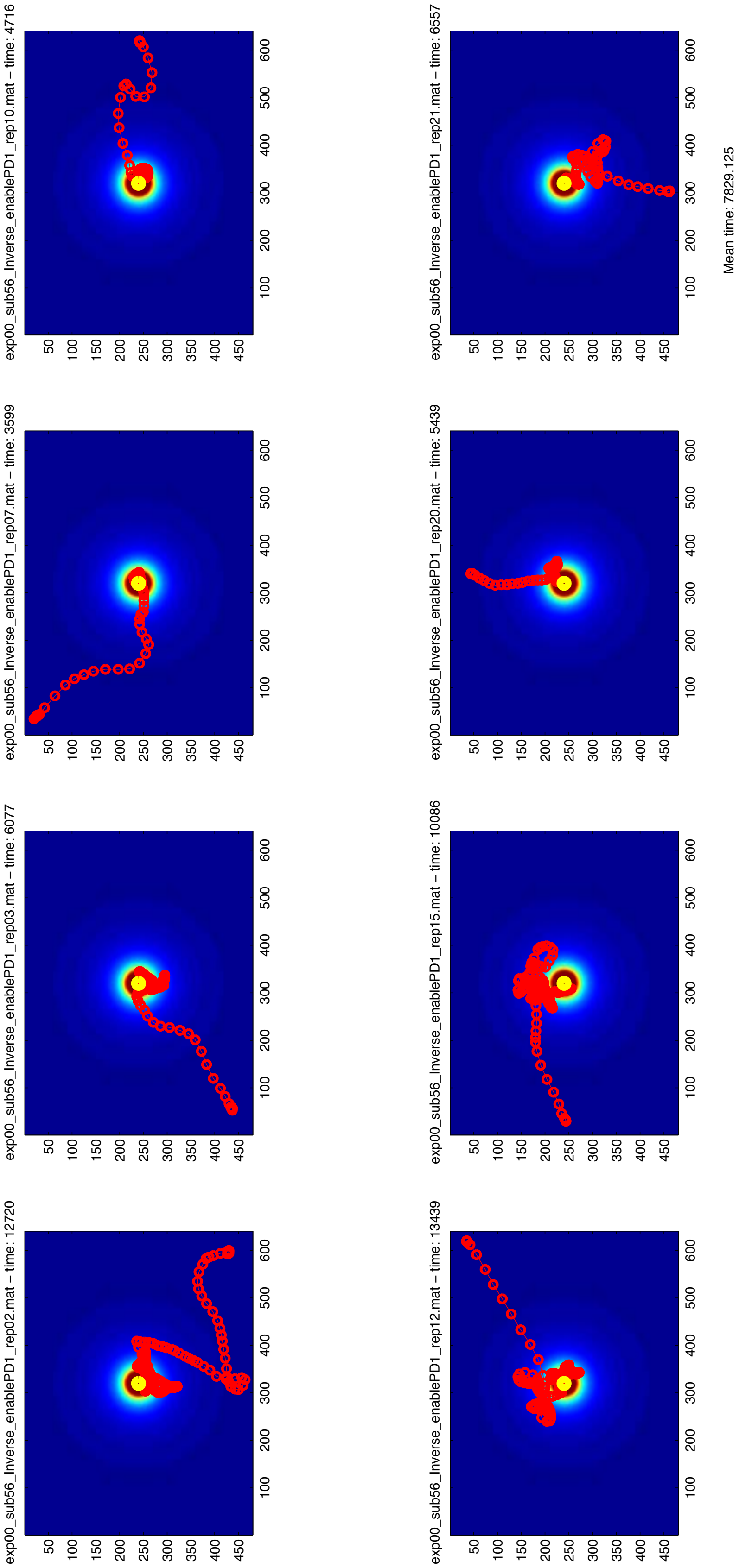
EXP 0 - ONLY TAMO



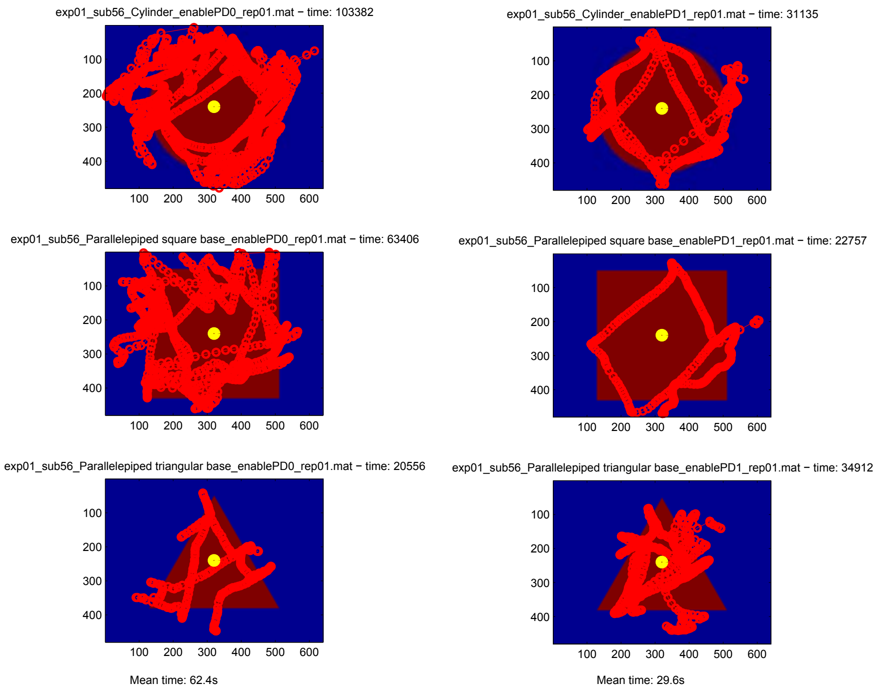
EXP 0 - ONLY AUDIO



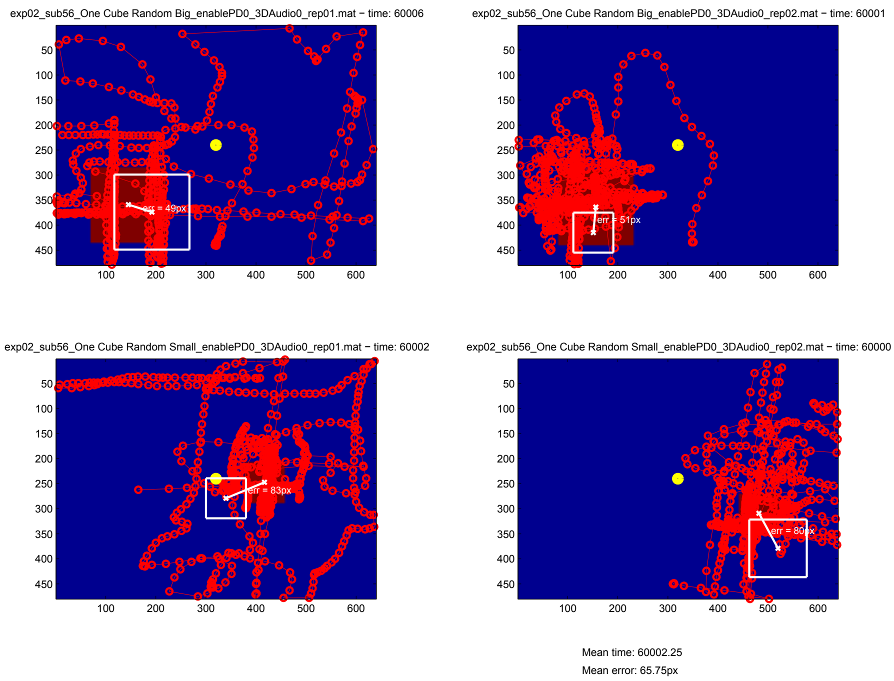
EXP 0 - TAMO AND AUDIO



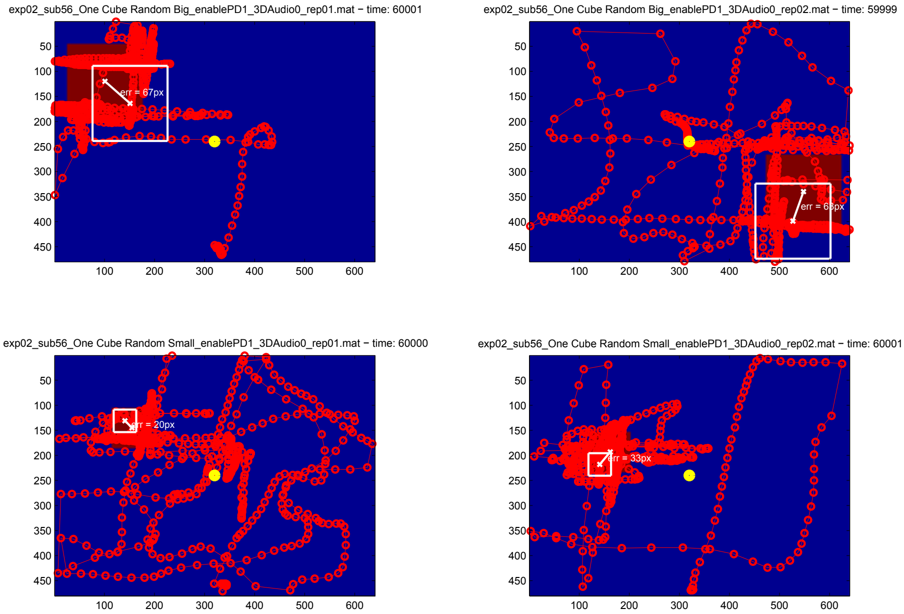
EXP 1 - NO AUDIO AGAINST AUDIO



EXP 2 - ONLY TAMO

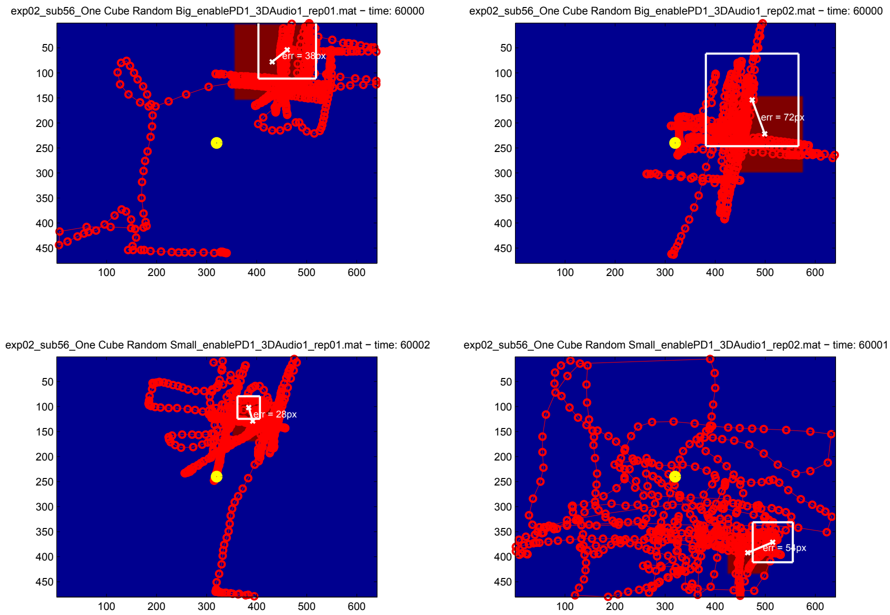


EXP 2 - TAMO AND AUDIO 2D



Mean time: 60000.25
Mean error: 47px

EXP 2 - TAMO AND AUDIO 3D



Mean time: 60000.75
Mean error: 48px

B

SOURCE CODE OF PSYCHOACOUSTIC

Source codes presented in this appendix have been dramatically shortened in order to be placed here. These codes work just as a reference and tutorial and it can not be used on real experiments. B.1 contains an adapted version of the source code of `experiment.m` that manages the psychoacoustic experiment while it is running. B.2 contains the algorithm already explained as pseudo-code in 4.3.

B.1 `experiment.m`

Listing 2: Experiment source code

```
function varargout = experiment(varargin)
% EXPERIMENT M-file for experiment.fig
% EXPERIMENT, by itself, creates a new EXPERIMENT or raises the existing
% singleton*.

gui_Singleton = 1;
gui_State = struct('gui_Name', mfilename, ...
                   'gui_Singleton', gui_Singleton, ...
                   'gui_OpeningFcn', @experiment_OpeningFcn, ...
                   'gui_OutputFcn', @experiment_OutputFcn, ...
                   'gui_LayoutFcn', [] , ...
                   'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before experiment is made visible.
function experiment_OpeningFcn(hObject, eventdata, handles, varargin)

% Enable PureData (debug purprose)
experimentConfigHandler = getappdata(0,'experimentConfigHandler');
handles.pureData = getappdata(experimentConfigHandler,'enablePureData');
```

```

% Trials order
handles.trialsOrder = getappdata(experimentConfigHandler,'trialsOrder');

% Part value
handles.part = getappdata(experimentConfigHandler,'part');

% Choose default command line output for experiment
handles.output = hObject;

% Azimuth angle
handles.azimuthRadius = 0.3;

% Update handles structure
guidata(hObject, handles);

% Import shared data (current subject ID)
subjectsHandler = getappdata(0,'subjectsHandler');
id = getappdata(subjectsHandler,'sharedID');
repetitions = getappdata(subjectsHandler,'repetitions');
handles.repetitions = repetitions;
pauseInterval = getappdata(subjectsHandler,'pauseInterval');
handles.pauseInterval = pauseInterval;

% Load subject list and extract data
subjectsStruct = load('subjects');
subjects = subjectsStruct.subjects;
handles.subjects = subjects;

if ( isempty(subjects(:,1)) ),
  disp('Database empty. Abort.');
  close();
end

numOfRecords = length(subjects(:,1));
for i = 1:numOfRecords,
  if ( subjects{i,6} == id )
    lastName = subjects{i,1};
    firstName = subjects{i,2};
    gender = subjects{i,3};
    age = subjects{i,4};
    [~, earDataFile, ~] = fileparts(subjects{i,5});
    earDataFile = [earDataFile '.mat'];
    cd img;
    earDataFile = load(earDataFile);
    cd ..;
    %disp(earDataFile);
    break;
  end
end

% Extract stored F0 value from ear trace.
F0 = earDataFile.F0;
```

```
clear earDataFile.F0;
%disp(F0);

% Load data from CIPIC Database and calculate mismatch values
handles.cipicIDs = listCipicIDs();
handles.numCipicSubjects = size(handles.cipicIDs,2);
numOfTests = 3;

sumMismatchCriteria_1 = calculateWeightedMismatch(id, F0, 1/3, 1/3, 1/3, 1);
sumMismatchCriteria_2 = calculateWeightedMismatch(id, F0, 1, 0, 0, 0);

% Before sorting we choose Kemar (CIPIC ID=21, idNum = 12) as HRTF.
sumMismatch = [sumMismatchCriteria_1(12,1) 12];

sumMismatchCriteria_1 = sortrows(sumMismatchCriteria_1);
sumMismatchCriteria_2 = sortrows(sumMismatchCriteria_2);

% Now we choose the best match HRTF from both criteria, avoiding
% duplicates
for i=1:handles.numCipicSubjects
  if ( ~ismember(sumMismatchCriteria_1(i,2), sumMismatch) )
    sumMismatch = [sumMismatch; sumMismatchCriteria_1(i,1) sumMismatchCriteria_1(i,2)];
    break;
  end
end

for i=1:handles.numCipicSubjects
  if ( ~ismember(sumMismatchCriteria_2(i,2), sumMismatch) )
    sumMismatch = [sumMismatch; sumMismatchCriteria_2(i,1) sumMismatchCriteria_2(i,2)];
    break;
  end
end

% IMPORTANT: on sumMismatch the first line is Kemar, second line about
% the first criteria and the third line about the second criteria.
disp('Mismatch - IDNUM')
disp(sumMismatch);

% We add a new column, as required for the next step. This column is
% used on output results in order to find out the match criteria used to
% select a particular HRTF.
newColumn = zeros(3,1);
newColumn(:,1) = 1:3;
```

```
    sumMismatch = [sumMismatch newColumn];

    idStr = handles.cipicIDs(sumMismatch(1:numOfTests,2)); % convert idNum to CIPIC IDs
    handles.idHrtfCIPIC = zeros(0);
    for i = 1:numOfTests
      handles.idHrtfCIPIC(end+1) = str2num(idStr{i});
    end
    handles.idHrtfCIPICMatching = (sumMismatch(1:numOfTests,3))';
    disp(handles.idHrtfCIPIC);
    handles.toBeLoadedCipicIDs = handles.idHrtfCIPIC;


    % Prepare arrays with experiment data, calculate test number and random
    % shuffle test order.

    choosenIDs = handles.idHrtfCIPIC;
    choosenIDsMatching = handles.idHrtfCIPICMatching;
    load('positions.mat');
    [numPositions, ~]  = size(positions);

    % Sort the trials according to the required order.
    % Stimulous are divided into blocks. Each block is created using
    % 1 HRTF and all positions. Each block is repeated 2 times.
    % Calling block with capital letters, possible orders are:
    %
    % Latin square 1: ABC BCA
    % Latin square 2: BCA CAB
    % Latin square 3: CAB ABC
    % Totally random: no blocks.
    %
    % The two same letters on the same row have a different position permutation.
    % In the following lines, A is the first row of the matrix, B the
    % second and so on...

    handles.numOfTests = 3 * numPositions * 2;
    blockSize = numPositions;
    block_A_idHrtfCIPIC = zeros(1, blockSize) + choosenIDs(1);
    block_A_idHrtfCIPICMatching = zeros(1, blockSize) + choosenIDsMatching(1);
    block_A_thetaHrtfCIPIC = positions(:,1)';
    block_A_phiHrtfCIPIC = positions(:,2)';
    block_B_idHrtfCIPIC = zeros(1, blockSize) + choosenIDs(2);
    block_B_idHrtfCIPICMatching = zeros(1, blockSize) + choosenIDsMatching(2);
    block_B_thetaHrtfCIPIC = positions(:,1)';
    block_B_phiHrtfCIPIC = positions(:,2)';
```

```
block_C_idHrtfCIPIC = zeros(1, blockSize) + choosenIDs(3);
block_C_idHrtfCIPICMatching = zeros(1, blockSize) + choosenIDsMatching(3);
block_C_thetaHrtfCIPIC = positions(:,1)';
block_C_phiHrtfCIPIC = positions(:,2)';
permutations_1 = randperm(blockSize);
permutations_2 = randperm(blockSize);
permutations_3 = randperm(blockSize);
permutations_4 = randperm(blockSize);
permutations_5 = randperm(blockSize);
permutations_6 = randperm(blockSize);

if ( strcmpi(handles.trialsOrder, 'latinSquare1') )
  handles.idHrtfCIPIC = [block_A_idHrtfCIPIC(permutations_1) block_B_idHrtfCIPIC(permutations_2) block_C_idHrtfCIPIC(permutations_3) ...
    block_B_idHrtfCIPIC(permutations_4) block_C_idHrtfCIPIC(permutations_5) block_A_idHrtfCIPIC(permutations_6)];
  handles.idHrtfCIPICMatching = [block_A_idHrtfCIPICMatching(permutations_1) block_B_idHrtfCIPICMatching(permutations_2) block_C_idHrtfCIPICMatching(permutations_3) ...
    block_B_idHrtfCIPICMatching(permutations_4) block_C_idHrtfCIPICMatching(permutations_5) block_A_idHrtfCIPICMatching(permutations_6)];
  handles.thetaHrtfCIPIC = [block_A_thetaHrtfCIPIC(permutations_1) block_B_thetaHrtfCIPIC(permutations_2) block_C_thetaHrtfCIPIC(permutations_3) ...
    block_B_thetaHrtfCIPIC(permutations_4) block_C_thetaHrtfCIPIC(permutations_5) block_A_thetaHrtfCIPIC(permutations_6)];
  handles.phiHrtfCIPIC = [block_A_phiHrtfCIPIC(permutations_1) block_B_phiHrtfCIPIC(permutations_2) block_C_phiHrtfCIPIC(permutations_3) ...
    block_B_phiHrtfCIPIC(permutations_4) block_C_phiHrtfCIPIC(permutations_5) block_A_phiHrtfCIPIC(permutations_6)];
elseif ( strcmpi(handles.trialsOrder, 'latinSquare2') )
  handles.idHrtfCIPIC = [block_B_idHrtfCIPIC(permutations_1) block_C_idHrtfCIPIC(permutations_2) block_A_idHrtfCIPIC(permutations_3) ...
    block_C_idHrtfCIPIC(permutations_4) block_A_idHrtfCIPIC(permutations_5) block_B_idHrtfCIPIC(permutations_6)];
  handles.idHrtfCIPICMatching = [block_B_idHrtfCIPICMatching(permutations_1) block_C_idHrtfCIPICMatching(permutations_2) block_A_idHrtfCIPICMatching(permutations_3) ...
    block_C_idHrtfCIPICMatching(permutations_4) block_A_idHrtfCIPICMatching(permutations_5) block_B_idHrtfCIPICMatching(permutations_6)];
```

```
        handles.thetaHrtfCIPIC = [block_B_thetaHrtfCIPIC(permutations_1) block_C_thetaHrtfCIPIC(permutations_2) block_A_thetaHrtfCIPIC(permutations_3) ...
          block_C_thetaHrtfCIPIC(permutations_4) block_A_thetaHrtfCIPIC(permutations_5) block_B_thetaHrtfCIPIC(permutations_6)];
        handles.phiHrtfCIPIC = [block_B_phiHrtfCIPIC(permutations_1) block_C_phiHrtfCIPIC(permutations_2) block_A_phiHrtfCIPIC(permutations_3) ...
          block_C_phiHrtfCIPIC(permutations_4) block_A_phiHrtfCIPIC(permutations_5) block_B_phiHrtfCIPIC(permutations_6)];
      elseif ( strcmpi(handles.trialsOrder, 'latinSquare3') )
        handles.idHrtfCIPIC = [block_C_idHrtfCIPIC(permutations_1) block_A_idHrtfCIPIC(permutations_2) block_B_idHrtfCIPIC(permutations_3) ...
          block_A_idHrtfCIPIC(permutations_4) block_B_idHrtfCIPIC(permutations_5) block_C_idHrtfCIPIC(permutations_6)];
        handles.idHrtfCIPICMatching = [block_C_idHrtfCIPICMatching(permutations_1) block_A_idHrtfCIPICMatching(permutations_2) block_B_idHrtfCIPICMatching(permutations_3) ...
          block_A_idHrtfCIPICMatching(permutations_4) block_B_idHrtfCIPICMatching(permutations_5) block_C_idHrtfCIPICMatching(permutations_6)];
        handles.thetaHrtfCIPIC = [block_C_thetaHrtfCIPIC(permutations_1) block_A_thetaHrtfCIPIC(permutations_2) block_B_thetaHrtfCIPIC(permutations_3) ...
          block_A_thetaHrtfCIPIC(permutations_4) block_B_thetaHrtfCIPIC(permutations_5) block_C_thetaHrtfCIPIC(permutations_6)];
        handles.phiHrtfCIPIC = [block_C_phiHrtfCIPIC(permutations_1) block_A_phiHrtfCIPIC(permutations_2) block_B_phiHrtfCIPIC(permutations_3) ...
          block_A_phiHrtfCIPIC(permutations_4) block_B_phiHrtfCIPIC(permutations_5) block_C_phiHrtfCIPIC(permutations_6)];
      elseif ( strcmpi(handles.trialsOrder, 'randomOrder') )
        handles.numOfTests = length(choosenIDs) * numPositions * repetitions;
        handles.idHrtfCIPIC = zeros(1, handles.numOfTests);
        handles.thetaHrtfCIPIC = zeros(1, handles.numOfTests);
        handles.phiHrtfCIPIC = zeros(1, handles.numOfTests);
        handles.idHrtfCIPICmatching = zeros(1, handles.numOfTests);
        pos = 1;
        for i = 1:length(choosenIDs)
          for j = 1:numPositions
            for k = 1:repetitions
              handles.idHrtfCIPIC(pos) = choosenIDs(i);
```

```
            handles.idHrtfCIPICMatching(pos) = choosenIDsMatching(i);
            handles.thetaHrtfCIPIC(pos) = positions(j, 1);
            handles.phiHrtfCIPIC(pos) = positions(j, 2);
            pos = pos + 1;
          end
      end
    end

    randomOrder = randperm(handles.numOfTests);
    handles.idHrtfCIPIC = handles.idHrtfCIPIC(randomOrder);
    handles.thetaHrtfCIPIC = handles.thetaHrtfCIPIC(randomOrder);
    handles.phiHrtfCIPIC = handles.phiHrtfCIPIC(randomOrder);
    handles.idHrtfCIPICMatching = handles.idHrtfCIPICMatching(randomOrder);
end

finalMatrix = [handles.idHrtfCIPIC; handles.idHrtfCIPICMatching; handles.thetaHrtfCIPIC; handles.phiHrtfCIPIC];

% Set default values
handles.currentTest = 0;
handles.id = id;
handles.chosenExternalization = 0;
handles.chosenElevation = 0;
handles.chosenConfidence = 0;
handles.alreadyChosenAzimuth = 0;
% HRTFid, HRTFmatching, thetaChosen, thetaReal, PhiChosen, PhiReal, ext, dist, conf, delay
handles.experimentResults = cell(handles.numOfTests, 10);

% Display subject info
handles.subjectData = [lastName ' ' firstName ' ' gender ' ' age ];
set(handles.subjectDataText, 'String', handles.subjectData);
set(handles.currentTestText, 'String', [int2str(handles.currentTest) ...
        ' of ' int2str(handles.numOfTests)]);

% Draw on axes
axes(handles.azimuthAxes);
img = imread('img/head_from_top.png');
imagesc([0, 1], [0, 1], flipdim(img,1));
set(gca,'YDir','normal');
rectHandle = rectangle('Position',[0.2,0.2,handles.azimuthRadius*2, ...
            handles.azimuthRadius*2],'Curvature',[1,1], 'LineWidth', 20);
set(rectHandle, 'edgecolor', [0.9, 0.9, 0.9]);

```

```
axes(handles.elevationAxes);
rectElevCenterHandle = rectangle('Position',[-0.3,-0.3,0.6,0.6],'Curvature',[1,1], 'LineWidth', 2);
set(rectElevCenterHandle, 'edgecolor', [0, 0, 0]);
rectElevHandle = rectangle('Position',[-1,-1,2,2],'Curvature',[1,1], 'LineWidth', 20);
set(rectElevHandle, 'edgecolor', [0.9, 0.9, 0.9]);
axis equal;
xlim([0 1.5]);
ylim([-1.5 1.5]);
hold on;
global elevationPointH;
elevationPointH = plot(1,0,'*', 'MarkerSize', 10);
set(gca,'xtick',[],'ytick',[]);

handles.hObject = hObject;
guidata(hObject, handles);
addlistener(handles.elevationSlider,'Value','PostSet', ...
@(varargin) elevationMoveCallback(varargin{:}, handles));

% Start Pure Data and network socket
if ( handles.pureData )
  startPD;
  addpath('../mex');
  pnet('closeall');
  handles.socket = pnet('udpsocket', 3455);
  pnet(handles.socket, 'udpconnect', '127.0.0.1', 3456);

  % Load HRTF
  waitBarHandler = waitbar(0/100,'Loading. Wait until you hear 3 impulse trains, ...
    even if the loading process is completed.','CreateCancelBtn', ...
    'setappdata(gcbf,''canceling'',1)');
  set(waitBarHandler,'Name','Loading HRTF');
  setappdata(waitBarHandler,'canceling',0)
  handles.waitBarHandler = waitBarHandler;
  guidata(hObject, handles);
  pause on;
  set(hObject, 'Visible', 'on');
  set(waitBarHandler, 'Visible', 'on');
  counter = 1;
  % 20s to start PD and 20s to load each HRTF (total time = 80s)
  for i = 1:800 % Loading time amount
    pause(0.1);
    if getappdata(waitBarHandler,'canceling')
      break;
    end;
    set(waitBarHandler, 'Visible', 'on');
    if ( i == 200 || i == 400 || i == 600)
      if ( handles.pureData )
        %disp(handles.toBeLoadedCipicIDs(counter));
```

```
            pd_packet = [single(handles.toBeLoadedCipicIDs(
                counter)) single(0) single(0)];
            [~, datastring]=cstruct(pd_packet);
            [~, zerostring]=cstruct(single(0));
            % OSC coding
            string=char(['/pd_packet' zerostring(1:2) ',
                ffffffff' zerostring(1:2) datastring]);
            pnet(handles.socket, 'write', string);
            pnet(handles.socket, 'writepacket');
            counter = counter + 1;
          end
        end;
        waitBarHandler = waitbar(i/800, waitBarHandler);
      end;
    end;
    pause(0.5);


    set(handles.listenButton,'Enable','on');
    set(hObject, 'Visible', 'on');

    % Debug purpose
    assignin('base','handlesExperiment',handles);

    if ( handles.pureData )
      delete(waitBarHandler);
    end;
    guidata(hObject, handles);

    varargout{1} = handles.output;

    % --- Executes during object creation, after setting all
         properties.
    function subjectDataText_CreateFcn(hObject, eventdata,
        handles)
    if ispc && isequal(get(hObject,'BackgroundColor'), get
        (0,'defaultUicontrolBackgroundColor'))
      set(hObject,'BackgroundColor','white');
    end

    % --- Executes during object creation, after setting all
         properties.
    function statusText_CreateFcn(hObject, eventdata,
        handles)
    if ispc && isequal(get(hObject,'BackgroundColor'), get
        (0,'defaultUicontrolBackgroundColor'))
      set(hObject,'BackgroundColor','white');
    end

    % --- Executes on button press in listenButton.
    function listenButton_Callback(hObject, eventdata,
        handles)
    if ( handles.currentTest == 0 )
      if ( strcmpi(handles.trialsOrder, 'latinSquare1') ||
          strcmpi(handles.trialsOrder, 'latinSquare2') ||
          strcmpi(handles.trialsOrder, 'latinSquare3') )
```

```
            if (handles.repetitions == 1 && handles.part == 1) % First half of the experiment
                handles.numOfTests = handles.numOfTests / 2;
            end
            if (handles.repetitions == 1 && handles.part == 2) % Second half of the experiment
                handles.currentTest = handles.numOfTests / 2;
                handles.numOfTests = handles.numOfTests;
            end
        end
        disp(handles.currentTest);
        disp(handles.numOfTests);
    end

    if ( (handles.part == 1 && handles.currentTest ~= 0) || (handles.part == 2 && handles.currentTest ~= handles.numOfTests / 2) )
        handles.endTime = round(etime(clock,handles.startTime) * 1000);
        %disp(handles.chosenAzimuth);
        %disp(handles.chosenElevation);
        %disp(handles.chosenExternalization);
        %disp(handles.chosenDistance);
        %disp(handles.chosenConfidence);
        %disp(handles.endTime);

        % We need to convert azimuth-elevation in order to match the format:
        % azimuth = -[0 180] on the left and azimuth = [0 180] on the right
        % elevation = [-90 90]
        currAzimuth = handles.thetaHrtfCIPIC(handles.currentTest);
        currElevation = handles.phiHrtfCIPIC(handles.currentTest);
        if ( (currElevation > 90) || (currElevation < -90) ) % change elevation
            if ( (currElevation > 90 && currElevation < 270) )
                handles.phiHrtfCIPIC(handles.currentTest) = 180 - handles.phiHrtfCIPIC(handles.currentTest);
            end
            if ( (currElevation < -90 && currElevation > -270) )
                handles.phiHrtfCIPIC(handles.currentTest) = -(180 + handles.phiHrtfCIPIC(handles.currentTest));
            end
            if ( (currElevation < -270 ) )
                handles.phiHrtfCIPIC(handles.currentTest) = 360 - handles.phiHrtfCIPIC(handles.currentTest);
            end
            if ( (currElevation > 270 ) )
                handles.phiHrtfCIPIC(handles.currentTest) = handles.phiHrtfCIPIC(handles.currentTest) - 360;
            end
        end
```

```
    if ( (currElevation > 90 && currElevation < 270) || ( currElevation < -90 && currElevation > -270) ) % change azimuth
      if ( currAzimuth >= 0 )
        handles.thetaHrtfCIPIC(handles.currentTest) = handles.thetaHrtfCIPIC(handles.currentTest) - 180;
      end
      if ( currAzimuth < 0 )
        handles.thetaHrtfCIPIC(handles.currentTest) = handles.thetaHrtfCIPIC(handles.currentTest) + 180;
      end
    end

    handles.experimentResults{handles.currentTest, 1} = handles.idHrtfCIPIC(handles.currentTest);
    handles.experimentResults{handles.currentTest, 2} = handles.idHrtfCIPICMatching(handles.currentTest);
    handles.experimentResults{handles.currentTest, 3} = handles.chosenAzimuth;
    handles.experimentResults{handles.currentTest, 4} = handles.thetaHrtfCIPIC(handles.currentTest);
    handles.experimentResults{handles.currentTest, 5} = handles.chosenElevation;
    handles.experimentResults{handles.currentTest, 6} = handles.phiHrtfCIPIC(handles.currentTest);
    handles.experimentResults{handles.currentTest, 7} = handles.chosenExternalization;
    handles.experimentResults{handles.currentTest, 8} = handles.chosenDistance;
    handles.experimentResults{handles.currentTest, 9} = handles.chosenConfidence;
    handles.experimentResults{handles.currentTest, 10} = handles.endTime;
  end

  handles.currentTest = handles.currentTest + 1;
  set(handles.currentTestText, 'String', [int2str(handles.currentTest) ' of ' int2str(handles.numOfTests)]);
  guidata(hObject, handles);

  if ( handles.currentTest > handles.numOfTests )
    cd results;
    results = handles.experimentResults;
    save([int2str(handles.id) '.mat'],'results');
    save([int2str(handles.id) '_part' num2str(handles.part) '.mat'],'results');
    cd ..;
    uiwait(msgbox('Experiment completed and saved.', ...
            'Completed','none'));
    guidata(hObject, handles);
    close();
    return;
  end

```

```
global elevationPointH;
% Reset GUI for next test
if ( (handles.part == 1 && handles.currentTest-1 ~= 0) ||
     (handles.part == 2 && handles.currentTest-1 ~= handles.numOfTests / 2) )
  set(handles.externalizationButtonGroup,'SelectedObject',[]);
  set(handles.listenButton,'Enable','off');
  set(handles.elevationSlider,'Enable','off');
  set(handles.externalizationOneRadioButton,'Enable','off');
  set(handles.externalizationTwoRadioButton,'Enable','off');
  handles.chosenElevation = 0;
  handles.chosenExternalization = 0;
  handles.chosenConfidence = 5;
  handles.alreadyChosenAzimuth = 0;
  delete(handles.azimuthPlotHandler);
else
  set(handles.externalizationButtonGroup,'SelectedObject',[]);
  set(handles.listenButton,'Enable','off');
  handles.chosenExternalization = 0;
  handles.chosenConfidence = 5;
end

% Pause and saving partial results
if ( handles.currentTest-1 ~= 0 && mod(handles.currentTest-1, handles.pauseInterval) == 0 )
  if ( (handles.part == 1) ||
     (handles.part == 2 && handles.currentTest-1 ~= handles.numOfTests / 2 ))
    cd results;
    results = handles.experimentResults;
    save([int2str(handles.id) '_autosave.mat'],'results');
    cd ..;
    countdownFig = countdown(0, 180);
    uiwait(countdownFig);
    box = msgbox('Click ok when ready: the next trial will start.', ...
            'Break finished!','help');
    uiwait(box);
  end
end

handles.startTime = clock;

set(handles.listenButton,'Enable','off');
set(handles.continueButton,'Enable','off');

pause(0.5);
if ( handles.pureData )
  pd_packet = [single(handles.idHrtfCIPIC(handles.currentTest)) single(handles.thetaHrtfCIPIC(
```

```
        handles.currentTest)) single(handles.phiHrtfCIPIC(handles.currentTest))];
    [~, datastring]=cstruct(pd_packet);
    [~, zerostring]=cstruct(single(0));
    % OSC coding
    string=char(['/pd_packet' zerostring(1:2) ',fffffffff' zerostring(1:2) datastring]);
    pnet(handles.socket, 'write', string);
    pnet(handles.socket, 'writepacket');
end;

set(handles.statusText,'String','Playing sound...');
pause(2);

set(handles.statusText,'String','Please choose sound elevation.');
set(handles.elevationSlider,'Value', 0);
set(handles.elevationSlider,'Enable','on');

guidata(hObject, handles);

% Debug purpose
assignin('base','handlesExperiment',handles);


% --- Executes on button press in continueButton.
function continueButton_Callback(hObject, eventdata, handles)
 guidata(hObject, handles);

% --- Executes when selected object is changed in externalizationButtonGroup.
function externalizationButtonGroup_SelectionChangeFcn(hObject, eventdata, handles)
switch get(get(handles.externalizationButtonGroup,'SelectedObject'),'Tag')
  case 'externalizationOneRadioButton', res = 1;
  case 'externalizationTwoRadioButton', res = 2;
  case 'externalizationThreeRadioButton', res = 3;
  case 'externalizationFourRadioButton', res = 4;
  case 'externalizationFiveRadioButton', res = 5;
end
handles.chosenExternalization = res;
guidata(hObject, handles);
if ( handles.chosenExternalization ~= 0 && handles.chosenConfidence ~= 0 )
    set(handles.listenButton,'Enable','on');
end



% --- Executes when selected object is changed in confidenceButtonGroup.
function confidenceButtonGroup_SelectionChangeFcn(hObject, eventdata, handles)
```

```
switch get(get(handles.confidenceButtonGroup,'SelectedObject'),'Tag')
  case 'confidenceOneRadioButton',  res = 1;
  case 'confidenceTwoRadioButton',  res = 2;
  case 'confidenceThreeRadioButton',  res = 3;
  case 'confidenceFourRadioButton',  res = 4;
  case 'confidenceFiveRadioButton',  res = 5;
end
handles.chosenConfidence = res;
guidata(hObject, handles);
if ( handles.chosenExternalization ~= 0 && handles.chosenConfidence ~= 0 )
    set(handles.listenButton,'Enable','on');
end


% --- Executes during object creation, after setting all properties.
function currentTestText_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
  set(hObject,'BackgroundColor','white');
end


% --- Executes when user attempts to close figure1.
function figure1_CloseRequestFcn(hObject, eventdata, handles)
if ( handles.currentTest > 1 && handles.currentTest <= handles.numOfTests )
  choice = questdlg('Save partial results?', ...
    'Save experiment', ...
    'Yes', 'No', 'Cancel', 'Cancel');
  switch choice
    case 'Yes'
      cd results;
      results = handles.experimentResults;
      save([int2str(handles.id) '.mat'],'results');
      cd ..;
    case 'No'

    case 'Cancel'
      return;
  end
end

% Close Pure Data
if ( handles.pureData )
  [~,~] = dos('taskkill /F /im pd.exe');
end;

delete(hObject);


```

```
% Return the angle (in degrees) between p1, p2, p3,
    where p1 is the vertex.
function output = getAngle(x1,y1, x2,y2, x3,y3)
 temp = atan2( abs((x2-x1)*(y3-y1)-(y2-y1)*(x3-x1)),
               (x2-x1)*(x3-x1)+(y2-y1)*(y3-y1) );
 output =  radtodeg(temp);



% --- Executes during object creation, after setting all
     properties.
function currentQuestionText_CreateFcn(hObject,
    eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get
    (0,'defaultUicontrolBackgroundColor'))
  set(hObject,'BackgroundColor','white');
end

% --- Executes on mouse press over figure background.
function figure1_ButtonDownFcn(hObject, eventdata,
    handles)
clickType = get(handles.figure1, 'SelectionType');
if strcmp(clickType, 'alt')
  if ( strcmpi(get(handles.listenButton,'Enable'), 'on')
      )
      listenButton_Callback(hObject, eventdata, handles)
  end
end


% --- Executes on slider movement.
function elevationSlider_Callback(hObject, eventdata,
    handles)
guidata(hObject, handles);
if ( handles.alreadyChosenAzimuth == 0 )
    chooseAzimuthOnPlot(hObject, eventdata, handles)
end


% --- Executes during object creation, after setting all
     properties.
function elevationSlider_CreateFcn(hObject, eventdata,
    handles)
if isequal(get(hObject,'BackgroundColor'), get(0,'
    defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function elevationMoveCallback(varargin)
global elevationPointH;
if ( exist('elevationPointH') )
  if ( ishandle(elevationPointH) )
    delete(elevationPointH);
  end
end
```

```
chosenElevation = get(varargin{end}.elevationSlider,'Value');
chosenElevation = asin(chosenElevation/90)/(pi/2)*90;
xElevation = cosd(chosenElevation);
yElevation = sind(chosenElevation);
elevationPointH = plot(varargin{end}.elevationAxes, xElevation, yElevation,'*', 'MarkerSize', 10);

function chooseAzimuthOnPlot(hObject, eventdata, handles)
set(handles.elevationSlider,'Enable','off');
set(handles.statusText,'String','Please choose sound source position.');
alreadyChosenAzimuth = false;

while ( true )
  try
    [xChosen,yChosen] = ginput(1);
  catch ME
    return
  end

  if ( xChosen < 0 || xChosen > 1 || yChosen < 0 || yChosen > 1 )
    uiwait(msgbox('Please choose a point inside the image.', 'Error','warn'));
    continue;
  end

  dist = pdist([xChosen yChosen; 0.5 0.5]);
  if ( dist > 0.5 || dist < 0.2 )
    uiwait(msgbox('Please choose a point near the circle.', 'Error','warn'));
    continue;
  end

  if ( gca == handles.azimuthAxes )
    xAzimuthChosen = xChosen;
    yAzimuthChosen = yChosen;
    hold on;
    if ( alreadyChosenAzimuth == true )
      azimuthPlotHandler = plot(xAzimuthChosen, yAzimuthChosen, '*');
    else
      azimuthPlotHandler = plot(xAzimuthChosen, yAzimuthChosen, '*');
      alreadyChosenAzimuth = true;
    end
  end
  if ( alreadyChosenAzimuth )
    break;
  end
end

handles.chosenAzimuth = getAngle(0.5, 0.5, 0.5, 1, xAzimuthChosen, yAzimuthChosen);
```

```
    if ( xAzimuthChosen < 0.5 )
        handles.chosenAzimuth = -handles.chosenAzimuth;
    end

    handles.chosenDistance = pdist([xAzimuthChosen yAzimuthChosen; 0.5 0.5]);
    handles.azimuthPlotHandler = azimuthPlotHandler;
    handles.alreadyChosenAzimuth = alreadyChosenAzimuth;

    guidata(hObject, handles);

    % Debug purpose
    assignin('base','handlesExperiment',handles);

    set(handles.externalizationOneRadioButton,'Enable','on');
    set(handles.externalizationTwoRadioButton,'Enable','on');
    set(handles.elevationSlider,'Enable','on');
    % set(handles.listenButton,'Enable','on'); will be triggered by
    % externalization and confidence callbacks.
```

B.2 calculatedweightedmismatch.m

Listing 3: Calculated Weighted Mismatch source code

```
function [ sumMismatch ] = calculateWeightedMismatch( subject, F0, weightOne, weightTwo, weightThree, printOutput )
%CALCULATEWEIGHTEDMISMATCH Calculates the mismatch between a given F0 and
%all NotchFreq stored in the CIPIC database.
%   the weight values refer to contour 1 (the most external), contour 2
%   (in the middle) and contour 3 (the most internal). They must be
%   inserted as percent values expressed in the interval [0, 1].
%   They should sum at 1.
%
%   You can use the printOutput variable in order to print the calculated
%   mismatch values on the Matlab Command Windows.

% Load data from CIPIC Database and calculate mismatch values for each
% contour
handles.cipicIDs = listCipicIDs();
handles.numCipicSubjects = size(handles.cipicIDs,2);

mismatch = zeros(handles.numCipicSubjects,3);
for idNum = 1:handles.numCipicSubjects
  cd cipic;
  load([’tracks’ handles.cipicIDs{idNum} ’.mat’], ’NotchFreq’);
  cd ..;
  for contour = 1:3
    % Some cipic subject do not have one (or more contours). We must
    % check for this and skip them.
    emptyContour = true;
    for phi = 1:17
      if ( NotchFreq(contour,phi) ~= 0 )
        emptyContour = false;
        break;
      end
    end
    if ( emptyContour )
      mismatch(idNum, contour) = -1;
      continue;
    end
    weight = 0;
    for phi = 1:17
    if ( NotchFreq(contour,phi) ~= 0 && F0(contour,phi) ~= 0 )
      mismatch(idNum, contour) = mismatch(idNum, contour) +
```

```
      abs(NotchFreq(contour,phi) - F0(contour,phi))/ NotchFreq(contour,phi);
      weight = weight + 1;
    end
    end
    mismatch(idNum, contour) = mismatch(idNum, contour) / weight;
  end
end

if ( printOutput )
  disp(mismatch);
end;

% Calculate the global mismatch using all avaiable contours according
% to the chosen weight.

weights = [0 0 0];
sumMismatch = zeros(handles.numCipicSubjects,2);
sumMismatch(:,2) = 1:handles.numCipicSubjects;
for idNum = 1:handles.numCipicSubjects
  % First of all, we calculate the number of avaiable mismatches
  % contours. We also remove the contours with required weight of
  % zero.
  if (weightOne == 0)
    mismatch(idNum, 1) = -1;
  end
  if (weightTwo == 0)
    mismatch(idNum, 2) = -1;
  end
  if (weightThree == 0)
    mismatch(idNum, 3) = -1;
  end
  avaiableContours = [];
  for contour = 1:3
    if ( mismatch(idNum, contour) ~= -1 )
      avaiableContours = [avaiableContours contour];
    end
  end
  % If there is only one contour, it get all the weight
  if ( length(avaiableContours) == 1 )
    weights = [0 0 0];
    weights(avaiableContours(1)) = 1;
  end
  % If there are two contours, they get half the weight of the other
  % one (we do not care if the other one is zero).
  if ( length(avaiableContours) == 2 )
    if ( ismember(1,avaiableContours) && ismember(2, avaiableContours) )
      localWeightOne = weightOne+weightThree/2;
      localWeightTwo = weightTwo+weightThree/2;
      localWeightThree = 0;
```

```
        end
        if ( ismember(1,avaiableContours) && ismember(3, avaiableContours) )
          localWeightOne = weightOne+weightTwo/2;
          localWeightThree = weightThree+weightTwo/2;
          localWeightTwo = 0;
        end
        if ( ismember(2,avaiableContours) && ismember(3, avaiableContours) )
          localWeightTwo = weightTwo+weightOne/2;
          localWeightThree = weightThree+weightOne/2;
          localWeightOne = 0;
        end
        weights = [localWeightOne localWeightTwo localWeightThree];
      end
      % If there are three contours, just use them!
      if ( length(avaiableContours) == 3 )
        weights = [weightOne weightTwo weightThree];
      end

      % Now we can calculate the weighted mean
      sumMismatch(idNum, 1) = mismatch(idNum, 1)*weights(1) + mismatch(idNum, 2)*weights(2) + mismatch(idNum, 3)*weights(3);

      % If sumMismatch is empty (-1), discard the value.
      if( sumMismatch(idNum, 1) == -1 )
        sumMismatch(idNum, 1) = 99;
      end
    end

    % If required, we can also print sumMismatch.
    % Here we are printing idNum, not cipidIDs, so:
    %
    % ID num | CIPIC ID
    % 1 -> 003
    % 2 -> 008
    % 3 -> 009
    % 4 -> 010
    % 5 -> 011
    % 6 -> 012
    % 7 -> 015
    % 8 -> 017
    % 9 -> 018
    % 10 -> 019
    % 11 -> 020
    % 12 -> 021
    % 13 -> 027
    % 14 -> 028
    % 15 -> 033
    % 16 -> 040
    % 17 -> 044
    % 18 -> 048
    % 19 -> 050
    % 20 -> 051
```

```
% 21 -> 058
% 22 -> 059
% 23 -> 060
% 24 -> 061
% 25 -> 065
% 26 -> 119
% 27 -> 124
% 28 -> 126
% 29 -> 127
% 30 -> 131
% 31 -> 133
% 32 -> 134
% 33 -> 135
% 34 -> 137
% 35 -> 147
% 36 -> 148
% 37 -> 152
% 38 -> 153
% 39 -> 154
% 40 -> 155
% 41 -> 156
% 42 -> 158
% 43 -> 162
% 44 -> 163
% 45 -> 165

sumMismatch = sumMismatch(sumMismatch(:,1)<99,:);

sortedSumMismatch = sortrows(sumMismatch);
if (~exist(['.\results\' int2str(subject) '.mat'], 'file
    '))
  data = 'emtpy'; %#ok<NASGU>
  save(['.\results\' int2str(subject) '.mat'],'data');
end
if (weightTwo == 0)
  sortedSumMismatch100 = sortedSumMismatch; %#ok<NASGU>
  save(['.\results\' int2str(subject) '.mat'],'
      sortedSumMismatch100', '-append');
else
  sortedSumMismatch13 = sortedSumMismatch; %#ok<NASGU>
  save(['.\results\' int2str(subject) '.mat'],'
      sortedSumMismatch13', '-append');
end

if ( printOutput )
  disp(sortedSumMismatch);
end

% Plot figure with mismatches
plotsHandle = figure;
set(gcf,'PaperPositionMode','auto')
set(plotsHandle, 'Position', [200 200 1200 300])
bar(sortedSumMismatch(:,1));
title(['Subject ' num2str(subject) ' mismatch values '
    num2str(weightOne) ' ' num2str(weightTwo) ' '
    num2str(weightThree)]);
```

```
set(gca,'XTick',1:45)
handles.cipicIDs = handles.cipicIDs(sortedSumMismatch(:
    ,2));
set(gca,'XTickLabel',handles.cipicIDs)
ylim([min(sortedSumMismatch(:,1))-min(sortedSumMismatch(
    :,1))/100*5, max(sortedSumMismatch(:,1))+max(
    sortedSumMismatch(:,1))/100*5]);
set(gca,'FontSize',8);
h=plotsHandle;
set(h,'PaperOrientation','landscape');
set(h,'PaperUnits','normalized');
set(h,'PaperPosition', [0 0 1 1]);
print(plotsHandle,'-dpdf', ['.\results\' int2str(subject
    ) '_mismatch_' num2str(weightOne) '_' num2str(
    weightTwo) '_' num2str(weightThree) '.pdf']);
close;

end
```

C SOURCE CODE OF MULTIMODAL

Source codes presented in this appendix have been dramatically shortened in order to be placed here. These codes work just as a reference and tutorial and it can not be used on real experiments. C.1 contains the code used in the generation of virtual environment maps and the code used in the real time experiment (TAMO and audio packets are sent to the COM port and as OSC UPD packets respectively).

C.1 startmapfigure.m

Listing 4: Start Map Figure source code

```
function completer = startMapFigure(inMap, inCipicID,
      inEnablePureData, inEnableTAMO, inConfiguration)
% inConfiguration: contains inConfiguration.executionTime that is the
% maximum allowed execution time. Set it to zero for disable maximum
% execution time. It also contains inConfiguration.exitDistance thas is the
% distance from the audio source in order to trigger the exit from the
% experimento. Set it to zero to disable this feature.

global A;
global refresher;
global explorationPath;
global iterations;
global cipicID;
global enablePureData;
global enableTAMO;
global serialPortHandler;
global start_byte;
global device_ID;
global executionTime;
global exitDistance;
global minDistanceForAzimuth;
global exitCounter;
global exitCounterSteps;
global soundIntensityScale;
global completerGlobal;
global soundAttenuationExpTwoValue;
global soundAttenuationExpTwo;
global enableAudio3D;
global tamoMaxElevation;
global wallSize;
global wallSwitch;

% Parse input parameters
```

```
cipicID = inCipicID;
enablePureData = inEnablePureData;
enableTAMO = inEnableTAMO;
enableAudio3D = inConfiguration.enableAudio3D;
executionTime = inConfiguration.executionTime;
exitDistance = inConfiguration.exitDistance;

% Configuration
refreshRateSeconds = 0.08;
serial_port = 'COM6';
start_byte = hex2dec('FF');
device_ID = hex2dec('0F');
exitCounterSteps = 15;
minDistanceForAzimuth = 5;
soundIntensityScale = 0.05;
soundAttenuationExpTwo = 1; %5 is a good value
tamoMaxElevation = 60;
wallSize = 0;

map.xMin = 1;
map.xMax = 640;
map.yMin = 1;
map.yMax = 480;
map.zMin = 60;
map.zMax = 120;
map.soundMarkerX = 320;
map.soundMarkerY = 240;
map.soundMarkerZ = 60;

% Auto config
soundAttenuationExpTwoValue = 1;
exitCounter = exitCounterSteps;
wallSwitch = 1;


% Start TAMO
if (enableTAMO)
  a = instrfind;
  delete(a)
  serialPortHandler = serial(serial_port);
  fopen(serialPortHandler);
  data = 90; % safe values between 40 90
  data_send = [start_byte device_ID data];
  fwrite(serialPortHandler, data_send);
end;

explorationPath.x = zeros(1,100);
explorationPath.y = zeros(1,100);
explorationPath.soundMarker = [map.soundMarkerX map.soundMarkerY map.soundMarkerZ];
iterations = 0;

% Generate Map
A = generateMap(A, inMap, map);
[mapFigure, mapAxes, ~] = makeMapFigure(A);
```

```
placeSoundMarker(mapAxes, map.soundMarkerX, map.soundMarkerY);
map.figure = mapFigure;

if ( enablePureData )
  socket = startAndConnectPD();
else
  socket = 0;
end;

% Start timers
completer = timer('Name', 'Completer', 'TimerFcn', @emptyCallback,...
              'ExecutionMode','singleShot','StartDelay', 100000);
start(completer);
completerGlobal = completer;
readyBox = msgbox('Click ok or press return when ready: the experiment will start.', ...
                   'Ready?','help');
  uiwait(readyBox);
  set (mapFigure, 'WindowButtonMotionFcn', @mouseMove);
  set(mapFigure,'KeyPressFcn',{@keyDownListener, mapFigure})
  refresher = timer('Name', 'Refresher', 'TimerFcn',{@refresherCallback, map, socket}, ...
          'ExecutionMode','fixedRate','Period', refreshRateSeconds);
  start(refresher);

end


function A = generateMap(A, inMap, map)
global soundAttenuationExpTwoValue;
global soundAttenuationExpTwo;
global wallSize;
global xCenter;
global yCenter;
switch inMap
  case 'Parallelepiped square base'
    A = createEmptyMap(map.xMin, map.xMax, map.yMin, map.yMax, map.zMin, map.zMax);
    A = createParallFromCenter(A, map.xMin, map.xMax, map.yMin, map.yMax, map.zMin,
         map.zMax, 320, 240, 380, 30);
  case 'Parallelepiped triangular base'
    A = createEmptyMap(map.xMin, map.xMax, map.yMin, map.yMax, map.zMin, map.zMax);
    A = createParall(A, map.xMin, map.xMax, map.yMin, map.yMax,
        map.zMin, map.zMax, [320 510 130], [50 380 380], 30);
  case 'Cylinder'
    A = createEmptyMap(map.xMin, map.xMax, map.yMin, map.yMax, map.zMin, map.zMax);
```

```
        A = createCylinder(A, map.xMin, map.xMax, map.yMin,
           map.yMax, map.zMin, map.zMax, 320, 240, 30, 190);
      case 'Random hills'
        A = createRandomHillsMap(map.xMin, map.xMax, map.
            yMin, map.yMax, map.zMin, map.zMax);
      case 'Noise'
        A = createNoise(map.xMin, map.xMax, map.yMin, map.
            yMax, map.zMin, map.zMax, 60, 120);
      case 'Inverse'
        A = createInverseDecMap(A, map.xMin, map.xMax, map.
            yMin, map.yMax,
           map.zMin, map.zMax, 320, 240, 2);
      case 'Horizontal Lines'
        A = createHorizLinesMap(map.xMin, map.xMax, map.yMin
            , map.yMax, map.zMin, map.zMax, 3);
      case 'One Cube Random Small'
        height = 30;
        cubeLength = 80;
        [xCenter, yCenter] = calculateSafeCubeCenterForHRTF(
            map, cubeLength, height);
        A = createEmptyMap(map.xMin, map.xMax, map.yMin, map
            .yMax, map.zMin, map.zMax);
        A = createParallFromCenter(A, map.xMin, map.xMax,
            map.yMin, map.yMax, map.zMin,
           map.zMax, xCenter, yCenter, cubeLength, height);
        soundAttenuationExpTwoValue = soundAttenuationExpTwo
            ;
      case 'One Cube Random Medium'
        height = 40;
        cubeLength = 115;
        [xCenter, yCenter] = calculateSafeCubeCenterForHRTF(
            map, cubeLength, height);
        A = createEmptyMap(map.xMin, map.xMax, map.yMin, map
            .yMax, map.zMin, map.zMax);
        A = createParallFromCenter(A, map.xMin, map.xMax,
            map.yMin, map.yMax, map.zMin,
           map.zMax, xCenter, yCenter, cubeLength, height);
        soundAttenuationExpTwoValue = soundAttenuationExpTwo
            ;
      case 'One Cube Random Big'
        height = 50;
        cubeLength = 150;
        [xCenter, yCenter] = calculateSafeCubeCenterForHRTF(
            map, cubeLength, height);
        A = createEmptyMap(map.xMin, map.xMax, map.yMin, map
            .yMax, map.zMin, map.zMax);
        A = createParallFromCenter(A, map.xMin, map.xMax,
            map.yMin, map.yMax, map.zMin,
          map.zMax, xCenter, yCenter, cubeLength, height);
        soundAttenuationExpTwoValue = soundAttenuationExpTwo
            ;
      otherwise
        msgbox('Invalid map chosen.', 'Invalid map','warn');
        return;
    end
    end
```

```

function [xCenter, yCenter] = calculateSafeCubeCenterForHRTF(map, cubeLength, height)
global wallSize;
while (true)
  xCenter = randInterval(map.xMin+cubeLength/2+wallSize, map.xMax-cubeLength/2-wallSize);
  yCenter = randInterval(map.yMin+cubeLength/2+wallSize, map.yMax-cubeLength/2-wallSize);
  distance = norm([xCenter yCenter] - [map.soundMarkerX map.soundMarkerY]);
  if ( distance < sqrt(2)*cubeLength/2 )
    continue;
  end
  distance = distance - sqrt(2)*cubeLength/2;
  if ( atand(height/distance) <= 45)
    break;
  end
end
end

function refresherCallback(~,~, map, socket)
global A;
global explorationPath;
global iterations;
global cipicID;
global enablePureData;
global enableTAMO;
global enableAudio3D;
global serialPortHandler;
global start_byte;
global device_ID;
global startTime;
global executionTime;
global exitDistance;
global timeStopper;
global exitCounter;
global exitCounterSteps;
global minDistanceForAzimuth;
global soundIntensityScale;
global tamoMaxElevation;
global wallSwitch;

mapWidth = size(A, 2);
mapHeight = size(A, 1);
cp = get(gca,'CurrentPoint');
id_x = round(cp(1,1));
id_y = round(cp(1,2));

if ( id_x >= 1 && id_x <= mapWidth && id_y >= 1 && id_y <= mapHeight )
  id_z = round(A(id_y, id_x));

  % calculate distance from the sound marker
```

```
    dist_2D = norm([id_x id_y] - [map.soundMarkerX map.
        soundMarkerY]);
    dist_3D = norm ([id_x id_y id_z] - [map.soundMarkerX
        map.soundMarkerY map.soundMarkerZ]);
    if ( enableAudio3D )
      intesity_scale_factor = inverseSquareLawScale(
          dist_3D, soundIntensityScale, map);
    else
      intesity_scale_factor = inverseSquareLawScale(
          dist_2D, soundIntensityScale, map);
    end

    % calculate angles
    azimuth = getAngle(id_x, id_y, id_x, id_y - 1 , map.
        soundMarkerX, map.soundMarkerY);
    if ( map.soundMarkerX < id_x )
      azimuth = -azimuth;
    end
    if ( id_z > map.soundMarkerZ )
      elevation = radtodeg(atan(dist_2D/(id_z-map.
          soundMarkerZ)))-90;
    else
      elevation = radtodeg(atan((map.soundMarkerZ-id_z)/
          dist_2D));
    end

    % If distance is very small, just set azimuth to zero
    if (dist_2D < minDistanceForAzimuth)
      azimuth = 0;
    end

    % send data to PD
    if ( enablePureData )
        pd_packet = [single(cipicID) single(azimuth)
            single(elevation*enableAudio3D) single(
            intesity_scale_factor)];
        [~, datastring]=cstruct(pd_packet);
        [~, zerostring]=cstruct(single(0));
        string=char(['/pd_packet' zerostring(1:2) ',
            ffffffff' zerostring(1:2) datastring]);
        pnet(socket, 'write', string);
        pnet(socket, 'writepacket');
    end;

    % send data to TAMO
    if (enableTAMO)
        data = round(A(id_y,id_x));
        data_send = [start_byte device_ID data];
        fwrite(serialPortHandler,data_send);
    end;

    % save position
    iterations = iterations + 1;
    explorationPath.x(iterations) = id_x;
    explorationPath.y(iterations) = id_y;

```

```
    %Start timers after the first feedback
    if ( iterations == 1 )
      if ( executionTime ~= 0 )
        timeStopper = timer('Name', 'timeStopper', 'TimerFcn',{@timeStopperCallback, map.figure},...
          'ExecutionMode','singleShot','StartDelay', executionTime);
        start(timeStopper);
      end
      startTime = clock;
    end

    % Exit if the pointer is near the sound source for at least
    % exitCounterSteps.
    if ( exitDistance > 0 && dist_2D <= exitDistance )
      if ( exitCounter == 0 )
        saveResults();
        closeExperiment(map.figure);
      else
        exitCounter = exitCounter - 1;
      end
    else
        exitCounter = exitCounterSteps;
    end
  else
    % We are out from the map. Set the TAMO to its max elevation as a
    % wall
    if (enableTAMO)
      wallSwitch = wallSwitch ~= 1;
      data = round(map.zMin+tamoMaxElevation-20*wallSwitch);
      data_send = [start_byte device_ID data];
      fwrite(serialPortHandler,data_send);
    end;
  end
  end


  function mouseMove (~, ~)
    % Do nothing. This function is needed just to update the CurrentPoint
    % location for pointer of the mouse/aptical device.
  end

  function emptyCallback (~, ~)
    % Do nothing. This function is needed just to trigger a timer
  end

  function [mapFigure, mapAxes, map] = makeMapFigure(A)
    mapFigure = figure('units','normalized','outerposition',[0.01 0.04 0.99 0.96]);
    set(mapFigure, 'menubar', 'none');
```

```
  set(mapFigure, 'CloseRequestFcn', {@closeMapFigureCallback, mapFigure});
  colordef none;
  colormap jet;
  set(mapFigure, 'Color', [0 0 0]);
  set(mapFigure, 'ToolBar', 'auto');
  mapAxes = axes('position', [0.01 0.02 0.96 0.96]);
  grid on
  box off
  map = imagesc(A, 'Parent', mapAxes);
  axis image;
end


function placeSoundMarker(mapAxes, soundMarkerX, soundMarkerY)
  axes(mapAxes); %#ok<MAXES>
  hold on;
  scatter(mapAxes, soundMarkerX, soundMarkerY, 'LineWidth', 6);
end


function A = createRandomHillsMap(xMin, xMax, yMin, yMax, zMin, zMax)
  xDim = xMax - xMin +1;
  yDim = yMax - yMin +1;
  A = zeros(yDim, xDim);

  xMin = xMin - round(xDim/2);
  yMin = yMin - round(yDim/2);
  xMax = xMax - round(xDim/2);
  yMax = yMax - round(yDim/2);
  x = linspace(xMin,xMax,xDim);
  y = linspace(yMin,yMax,yDim);
  [xx,yy] = meshgrid(x,y);

  for k = 1:20
      xc = (xMax - xMin)*rand(1) + xMin;
      yc = (yMax - yMin)*rand(1) + yMin;
      zz = exp( - ((xx-xc).^2 + (yy-yc).^2) / (min(xDim, yDim)*5));
      A = A +  zz;
  end

  A = A./max(max(A)) * (zMax-zMin) + zMin;
end


function A = createHorizLinesMap(xMin, xMax, yMin, yMax, zMin, zMax, lines)
  xDim = xMax - xMin +1;
  yDim = yMax - yMin +1;
  A = zeros(yDim, xDim);
  pixelsPerRow = floor(yDim/lines);
  pixelsLastRow = yDim - pixelsPerRow*(lines-1);
```

```
    zPerRow = round(linspace(zMax,zMin,lines));
    for i=0:lines-1
      if (i ~= lines-1)
        for j=1:pixelsPerRow
          A(i*pixelsPerRow+j,:) = A(i*pixelsPerRow+j,:) +
              zPerRow(i+1);
        end
      else
        for j=1:pixelsLastRow
          A(i*pixelsPerRow+j,:) = A(i*pixelsPerRow+j,:) +
              zPerRow(i+1);
        end
      end
    end
  end

  function A = createEmptyMap(xMin, xMax, yMin, yMax, zMin
      , zMax) %#ok<INUSD>
    xDim = xMax - xMin +1;
    yDim = yMax - yMin +1;
    A = zeros(yDim, xDim) + zMin;
  end

  function A = createEmptyMapWithWall(xMin, xMax, yMin,
      yMax, zMin, zMax, size) %#ok<INUSL>
    global tamoMaxElevation;
    xDim = xMax - xMin +1;
    yDim = yMax - yMin +1;
    A = zeros(yDim, xDim) + zMin;
    for x=1:size
      for y=1:yMax
        A(y,x) = zMin+tamoMaxElevation;
      end
    end
    for x=xMax-size+1:xMax
      for y=1:yMax
        A(y,x) = zMin+tamoMaxElevation;
      end
    end
    for y=1:size
      for x=1:xMax
        A(y,x) = zMin+tamoMaxElevation;
      end
    end
    for y=yMax-size+1:yMax
      for x=1:xMax
        A(y,x) = zMin+tamoMaxElevation;
      end
    end
  end

  function A = createParall(A, xMin, xMax, yMin, yMax,
      zMin, zMax, xv, yv, height)
    if (height > zMax - zMin)
      height = zMax - zMin;
    end
```

```
    [x, y] = meshgrid(xMin:xMax, yMin:yMax);
    IN = inpolygon(x,y,xv,yv);
    A = A .* (~IN);
    A = A + IN .* (zMin+height);
  end

  function A = createParallFromCenter(A, xMin, xMax, yMin,
       yMax, zMin, zMax, xCenter, yCenter, side, height)
    if (height > zMax - zMin)
      height = zMax - zMin;
    end
    xv =[xCenter-round(side/2), xCenter+round(side/2),
        xCenter+round(side/2) ,xCenter-round(side/2)];
    yv = [yCenter-round(side/2), yCenter-round(side/2),
        yCenter+round(side/2) ,yCenter+round(side/2)];
    A = createParall(A, xMin, xMax, yMin, yMax, zMin, zMax
        , xv, yv, height);
  end

  function A = createCylinder(A, xMin, xMax, yMin, yMax,
      zMin, zMax, xCenter, yCenter, height, radius)
    try
      A = load('./maps/cylinder.mat', 'cylinder');
      A = A.cylinder;
    catch err %#ok<NASGU>
      if (height > zMax - zMin)
        height = zMax - zMin;
      end
      xDim = xMax - xMin +1;
      yDim = yMax - yMin +1;
      IN = zeros(yDim, xDim);
      for y = 1:yDim
        for x = 1:xDim
          if ( pdist([xCenter yCenter; x y]) <= radius )
            IN(y,x) = 1;
          else
            IN(y,x) = 0;
          end
        end
      end
      A = A .* (~IN);
      A = A + IN .* (zMin+height);
      cylinder = A; %#ok<NASGU>
      save('./maps/cylinder.mat', 'cylinder');
    end
  end

  function A = createInverseDecMap(A, xMin, xMax, yMin,
      yMax, zMin, zMax, xCenter, yCenter, scaleFactor) %#
      ok<INUSL>
    global tamoMaxElevation;
    global exitDistance;
    xDim = xMax - xMin +1;
    yDim = yMax - yMin +1;
    A = zeros(yDim, xDim);
    for y = 1:yDim
```

```
      for x = 1:xDim
        distance= norm([xCenter yCenter] - [x y]);
        if ( distance <= exitDistance)
          A(y,x) = zMin + tamoMaxElevation;
        else
          A(y,x) = zMin + tamoMaxElevation*((exitDistance/distance)^scaleFactor);
        end
      end
    end
  end

  function A = createNoise(xMin, xMax, yMin, yMax, zMin, zMax, min, max)
    if (max - min > zMax - zMin)
        max = zMax;
        min = zMin;
    end
    xDim = xMax - xMin +1;
    yDim = yMax - yMin +1;
    A = zeros(yDim, xDim);
    for y = 1:yDim
      for x = 1:xDim
        A(y,x) = randInterval(min,max);
      end
    end
  end

  function A = createPiramid(A, xMin, xMax, yMin, yMax, zMin, zMax, xCenter, yCenter, height)
    if (height > zMax - zMin)
        height = zMax - zMin;
    end
    xDim = xMax - xMin +1;
    yDim = yMax - yMin +1;
    [x, y] = meshgrid(-xDim/2:xDim/2-1, -yDim/2:yDim/2-1);
    Z = (height/2-abs(x)) + (height/2-abs(y));
    Z(Z<0) = 0;
    Z = imrotate(Z,45,'crop');
    Z = circshift(Z,[yCenter-yDim/2 xCenter-xDim/2]);
    A = A + Z;
  end

  function A = createPiramids(A, xMin, xMax, yMin, yMax, zMin, zMax, height)
    try
      A = load('./maps/piramids.mat', 'piramids');
      A = A.piramids;
    catch err %#ok<NASGU>
      for i=0:7
        for j=0:5
          A = createPiramid(A, xMin, xMax, yMin, yMax, zMin, zMax, 40+80*i, 40+80*j, height);
          piramids = A; %#ok<NASGU>
          save('./maps/piramids.mat', 'piramids');
        end
```

```
      end
    end
  end

  function keyDownListener(~,~, mapFigure)
    saveResults();
    closeExperiment(mapFigure);
  end

  function closeMapFigureCallback(~,~, mapFigure)
    choice = questdlg('Save partial results?', ...
        'Save experiment', ...
        'Yes', 'No', 'Cancel', 'Cancel');
    switch choice
      case 'Yes'
          saveResults();
      case 'No'
          validResult = 0; %#ok<NASGU>
          save('./results/temp.mat', 'validResult');
      case 'Cancel'
          return;
    end
    closeExperiment(mapFigure);
  end

  function timeStopperCallback(~,~, mapFigure)
   saveResults();
   closeExperiment(mapFigure);
  end

  function saveResults()
    global explorationPath;
    global startTime;
    global iterations;
    global xCenter;
    global yCenter;
    global A; %#ok<NUSED>
    explorationPath.iterations = iterations;
    totalTime = round(etime(clock,startTime) * 1000); %#ok
        <NASGU>
    validResult = 1; %#ok<NASGU>
    if ( exist('xCenter', 'var') )
        xCenterCube = xCenter; %#ok<NASGU>
        yCenterCube = yCenter; %#ok<NASGU>
    end
    save('./results/temp.mat', 'A', 'explorationPath', '
        totalTime', 'validResult', 'xCenterCube', '
        yCenterCube');
  end

  function closeExperiment(mapFigure)
    global enablePureData;
    global refresher;
    global timeStopper;
    global executionTime;
    global iterations;
```

```
    global completerGlobal;
    stop(completerGlobal);
    if (executionTime ~= 0 && iterations > 1 )
        stop(timeStopper)
        delete(timeStopper)
    end
    stop(refresher)
    delete(refresher)
    delete(mapFigure)
    if ( enablePureData )
        [~,~] = dos('taskkill /F /im pd.exe');
    end;
    return;
  end

  function socket = startAndConnectPD()
    startPD;
    addpath('../mex');
    pnet('closeall');
    socket = pnet('udpsocket', 3455);
    pnet(socket, 'udpconnect', '127.0.0.1', 3456);
  end


  function angle = getAngle(x1,y1, x2,y2, x3,y3)
   temp = atan2( abs((x2-x1)*(y3-y1)-(y2-y1)*(x3-x1)), (x2-x1)*(x3-x1)+(y2-y1)*(y3-y1) );
   angle =  radtodeg(temp);
  end

  function scale_factor = inverseSquareLawScale(distance, min_scale, map)
    global A;
    global soundAttenuationExpTwoValue;
    distanceFromTopLeft = norm ([map.xMin map.yMin A(map.yMin, map.xMin)] - ...
     [map.soundMarkerX map.soundMarkerY map.soundMarkerZ]);
    distanceFromTopRight = norm ([map.xMax map.yMin A(map.yMin, map.xMax)] - ...
     [map.soundMarkerX map.soundMarkerY map.soundMarkerZ]);
    distanceFromBottomLeft = norm ([map.xMin map.yMax A(map.yMax, map.xMin)] - ...
     [map.soundMarkerX map.soundMarkerY map.soundMarkerZ]);
    distanceFromBottomRight = norm ([map.xMax map.yMax A(map.yMax, map.xMax)] - ...
     [map.soundMarkerX map.soundMarkerY map.soundMarkerZ]);
    max_distance = max([distanceFromTopLeft, distanceFromTopRight, ...
                  distanceFromBottomLeft, distanceFromBottomRight]);
    if (distance < 1)
      distance = 1;
```

```
    end
    distance_scale = (max_distance/distance)^2;
    min_scale = min_scale * soundAttenuationExpTwoValue;
    scale_factor = min_scale*distance_scale;
  end

  function num = randInterval(a,b)
    num = a + round((b-a).*rand());
  end
```

BIBLIOGRAPHY

[1] V. Algazi, R. Duda, D. Thompson, and C Avendano. The cipic hrtf database. In *In Proc. 2001 IEEE Workshop on Applications of Signal Processing to Audio and Electroacoustics*, 2001.

[2] Federico Avanzini and Giovanni De Poli. *Fundamentals of digital audio processing*, volume 1. Unpublished, 2012.

[3] D.W. Batteau. The role of the pinna in human localization. *Proc. R. Soc. London. Series B, Biological Sciences*, 168:158–160, 1967.

[4] D. Begault. *3D Sound for Virtual Reality and Multimedia*. Academic Press, 1994.

[5] D. Begault, E. Wenzel, and M. Anderson. Direct comparison of the impact of head tracking, reverberation, and individualized head-related transfer functions on the spatial perception of a virtual speech source. *J. Audio En*, 49(10):904–916, 2001.

[6] S. James Biggs and M. Srinivasan. *Haptic Interfaces in Handbook of Virtual Environment: Design, Implementation and Applications*. Massachusetts Institute of Technology, 2002.

[7] Jens Blauert. *The Psychophysics of Human Sound Localization*. Massachusetts Institute of Technology, 1997.

[8] Gianluca Boscolo Bariga. Management of heterogeneous hrtf databases for real time 3d audio rendering (Italian). Master's thesis, University of Padua, Department of Information Engineering, 2012.

[9] Doug A. Bowman, Joseph L. Gabbard, and Deborah Hix. A survey of usability evaluation in virtual environments: classification and comparison of methods. *Presence: Teleoper. Virtual Environ.*, 11:404–424, August 2002. ISSN 1054-7460. doi: 10.1162/105474602760204309.

[10] L. Brayda, C. Campus, and Monica Gori. What you touch is what you get: self-assessing a minimalist tactile sensory substitution device.

[11] Luca Brayda, Claudio Campus, Ryad Chellali, and Guido Rodriguez. Objective evaluation of spatial information acquisition using a visuo tactile sensory substitution device. In Shuzhi Ge, Haizhou Li, John John Cabibihan, and Yeow Tan, editors, *Social Robotics*, volume 6414 of *Lecture Notes in Computer Science*, pages 315–324. Springer Berlin Heidelberg, 2010.

[12] Luca Brayda, Claudio Campus, Ryad Chellali, Guido Rodriguez, and Cristina Martinoli. An investigation of search behaviour

in a tactile exploration task for sighted and non-sighted adults. In *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems*, CHI EA 2011, pages 2317–2322, New York NY, USA, 2011. ACM.

[13] A. Bujnowsky M. Drozd, R. Kowalik, and J. Wtorek. A tactile system for informing the blind on direction of a walk. In *Conference on Human System Interaction*, 2008.

[14] C. Campus, L. Brayda, F. De Carli, R. Chellali, F. Famà, C. Bruzzo, L. Lucagrossi, and G. Rodriguez. Tactile exploration of virtual objects for blind and sighted people: the role of beta 1 eeg band in sensory substitution and supramodal mental mapping. *Journal of Neurophysiology*, 107(10):2713–2729, 2012.

[15] R. Chellali, L. Brayda, C. Martinoli, and E Fontaine. How taxel based displaying devices can help visually impaired people to navigate safely. In *Autonomous Robots and Agents ICARA 4th International Conference on*, pages 470–475, 2009.

[16] C.I. Cheng and G.H. Wakefield. Introduction to head-related transfer functions (hrtfs): Representations of hrtfs in time, frequency, and space. *J. Audio Eng. Soc.*, 52:231–249, 2001.

[17] Bryan WC Chung. *Multimedia Programming with Pure Data*. Packt, 2013.

[18] S. Brungart Douglas, D. Romigh Griffin, and D. Simpson Brian. Head-related transfer function enhancement for improved vertical-polar localization. *J. Acoust. Soc. Am.*, 125(4):2691–2691, 2009.

[19] David Doukhan and Anne Sédès. Cw binaural: A binaural synthesis external for pure data. In *PDCon09*, 2009.

[20] R. Farcy, R. Leroux, R. A. Jucha, C. Grgoire, and A. Zogaghi. Electronic travel aids and electronic orientation aids for blind people: Technical, rehabilitation and everyday life points of view. In *Proceedings in CVHI 2006, The 4th european Conference and workshop on assistive technologies for Vision and Hearing Impairement*, 2006.

[21] W.G. Gardner and K.D. Martin. Hrtf measurements of a kemar. *J. Acoust. Soc. Am.*, 97:3907–3908, 1995.

[22] M. Geronazzo, S. Spagnol, and F. Avanzini. Estimation and modeling ofpinna-related transfer functions. In *Proc. 13th Int. Conf. Digital Audio Effects, Graz, Austria*, 2010.

[23] Stuart Goose and Carsten Moller. A 3d audio only interactive web browser: Using spatialization to convey hypermedia document structure. In *In Proceedings of the seventh ACM international conference of Multimedia (Part 1)*, pages 363–371, 1999.

[24] Massimo Grassi and Alessandro Soranzo. Mlp: A matlab toolbox for rapid and reliable auditory threshold estimation. *Behavior Research Methods*, 41:20–28, 2009.

[25] David M. Green. A maximum-likelihood method for estimating thresholds in a yes-no task. *J.Acoust Soc.Am*, 93:2096–2105, 1992.

[26] Deborah Hix and H. Rex Hartson. *Developing User Interfaces: Ensuring Usability Through Product & Process*. Wiley, 1993.

[27] Nicholas P. Holmes and Charles Spence. Multisensory integration: Space, time, & superadditivity. *Europe PMC Funders Group*, 15: 762–764, 2006.

[28] Sungmok Hwang, Youngjin Park, and Youn-sik Park. Modeling and customization of head-related impulse responses based on general basis functions in time domain. *Acta Acustica United with Acustica*, 94:965–980, 2008.

[29] Alejandro Jaimes and Nicu Sebe. Multimodal human-computer interaction: A survey. *Comput. Vis. Image Underst.*, 108:116–134, October 2007. ISSN 1077-3142. doi: 10.1016/j.cviu.2006.10.019.

[30] J. Jot and O. Warusfel. A spatial processor for musicians and sound engineers. In *CIARM: International Conference on Acoustics and Musical Research, Ferrara*, 1995.

[31] Alan Kan, Craig Kin, and Andre van Schaik. A psychophysical evaluation of near-field head-related transfer functions synthesized using a distance variation function. *J. Acoust. Soc. Am.*, 125(4):2233–2242, 2009.

[32] B. F. Katz and Gaetan Parseihian. Perceptually based head-related transfer function database optimization. *J. Acoust. Soc. Am.*, 131(2): 99–105, 2012.

[33] D. J. Kistler and F. L. Wightman. A model of head-related transfer functions based on principal components analysis and minimum-phase reconstruction. *J.*, 91(3):1637–1647, 1992.

[34] Orly Lahav and David Mioduser. Exploration of unknown spaces by people who are blind using a multi-sensory virtual environment. *Journal of Special Education Technology*, 19(3):15–23, 2004.

[35] Orly Lahav and David Mioduser. Construction of cognitive maps of unknown sspace using a multi-sensory virtual environment for people who are blind. *Computers in Human Behavior*, 24:1139–1155, 2008.

[36] A. Lecuyer, P. Mobuchon, C. Megard, J. Perret, C. Andriot, and J.-P. Colinot. Homere: a multimodal system for visually impaired people to explore virtual environments. In *Virtual Reality, 2003. Proceedings. IEEE*, pages 251–258, 2003. doi: 10.1109/VR.2003.1191147.

[37] Susan J Lederman and Roberta L Klatzky. Hand movements: A window into haptic object recognition. *Cognitive Psychology*, 19 (3):342 – 368, 1987. ISSN 0010-0285. doi: 10.1016/0010-0285(87) 90008-9.

[38] E.A. Lopez-Poveda and R. Meddis. A physical model of sound diffraction and reflections in the human concha. *J. Acoust. Soc. Am.*, 100(5):3248–3259, 1996.

[39] L Magee and J. Kennedy. Explblind pictures tactually. *Nature (London, United Kingdom)*, 283:287–288, 1980.

[40] Robert Mandl. Orthogonal latin squares: an application of experiment design to compiler testing. *Communications of the ACM*, 28: 1054–158, 1985.

[41] R. J. McAulay and T. F. Quatieri. S[eech analysis/synthesis based on a sinusoidal representation. *IEEE Trans. Acoust., Speech, Signal Process*, 34:744–754, 1986.

[42] M. Alex Meredith, James W. Nemitz, and Barry E. Stein. Determinants of multisensory integration in superior colliculus neurons. i. temporal factors. *The Journal of Neuroscience*, 7:3215–3229, 1987.

[43] P. Minnaar, J. Plogsties, S. Olesen, F. Christensen, and H. Meller. The interaural time difference in binaural synthesis. *Audio Engineering Society*, 2000.

[44] P. Mokhtari, H. Takemoto, R. Nishimura, and H. Kato. Acoustic simulation of kemar's hrtfs: Verification with measurements and the effects of modifying head shape and pinna concavity. In *Proc. Int. Work. Princ. Appl. Spatial Hearing (IWPASH), Zao, Miyagi, Japan,*, 2009.

[45] B. C. J. Moore, S. R. Oldfield, and G. J. Dooley. Detection and discrimination of spectral peaks and notches at 1 and 8 khz. *J*, 56: 1829–1834, 1989.

[46] Henrik Møller, Dorte Hammershøi, Clemen Boje Jensen, and Michael Friis Sørensen. Transfer characteristics of headphones measured on human ears. *J. Audio Eng. Soc*, 43:203–217, 1995.

[47] A. Nahvi, D.D. Helson, J. Hollerbach, and D.E. Johnson. Haptic manipulation of virtual mechanisms from mechanical cad designs. *Proc. IEEE Intl. Conf. Robotics & Automation, Leuven, Belgium*, 1: 375–380, 1998.

[48] A. Pentland. Looking at people. *Comm. of the ACM*, 43(3):35–44, 2000.

[49] S. Perret and W. Noble. The effect of head rotations on vertical plane sound localization. *J. Acoust. Soc. Amer.*, 102:2325–2332, 1997.

[50] V.C. Raykar, R. Duraiswami, and B. Yegnanarayana. Extracting the frequencies of the pinna spectral notches in measured head related impulse responses. *J. Acoust. Soc. Am.*, 118(1):364–374, 2005.

[51] D.W. Schloerb, O. Lahav, J.G. Desloge, and M.A. Srinivasan. Blindaid: Virtual environment system for self-reliant trip planning and orientation and mobility training. In *Haptics Symposium, 2010 IEEE*, pages 363 –370, march 2010. doi: 10.1109/HAPTIC.2010.5444631.

[52] Bernhard U. Seeber and Hugo Fastl. Subjective selection of non-individual head-related transfer functions. *Proceedings of the 2003 International Conference on Auditory Display*, 3:259–262, 2003.

[53] Ki Hoon Shin. Enhanced vertical perception through head-related impulse response customization based on pinna response tuning in the median plane. *IEICE trans, fundamentals*, E91-A:345–355, 2008.

[54] Ben Shneiderman, Catherine Plaisant, Maxine S. Cohen, and Steven M. Jacobs. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Prentice Hall, 2009.

[55] M. Siegel. Tactile display development: the driving force for tactile sensor development. *IEEE International Workshop*, 1:115–118, 2002.

[56] Simone Spagnol, Michele Geronazzo, and Federico Avanzini. Hearing distance: A low-cost model for near-field binaural effects. In *20th European Signal Processing Conference*, pages 2005–2009, 2012.

[57] Simone Spagnol, Michele Geronazzo, and Federico Avanzini. On the relation between pinna reflection patterns and head-related transfer function features. *IEEE transactions on audio, speech, and language processing*, 21:508–519, 2013.

[58] H. Takemoto, P. Mokhtari, H. Kato, R. Nishimura, and K. Iida. A simple pinna model for generating head-related transfer functions in the median plane. In *Proc. 20th Int. Congr. Acoust. (ICA 2010), Sydney, Australia*, 2010.

[59] R. Teranishi and E. A. B. Shaw. External-ear acoustic model with simple geometry. *J. Acoust. Soc. Am.*, 44:257–263, 1968.

[60] I. Viaud-Delmon, F. Znaidi, N. Bonneel, D. Doukhan, C. Suied, O. Warusfel, K V N Guyen, and G. Drettakis. Auditory-visual virtual environments to treat dog phobia. In *International Conference Series On Disability, Virtual Reality And Associated Technologies (ICDVRAT)*, 2008.

[61] Bruce N. Walker and Jeffrey Lindsay. Navigation performance with a virtual auditory display: Effects of beacon sound, capture

radius, and practice. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 48(2):265–278, Summer 2006. doi: 10.1518/001872006777724507.

[62] Matthew Wright, Adrian Freed, and Ali Momeni. Open sound control: State of the art 2003. In *International Conference on New Interfaces for Musical Expression*, 2003.

[63] V.P. Zinchenko and B.F. Lomov. The functions of hand and eye movements in the process of perception. *Problems of Psychology*, 1:12–26, 1960.

[64] D.N. Zotkin, R. Duraiswami, and L.S. Davis. Rendering localized spatial audio in a virtual auditory space. *Multimedia, IEEE Transactions on*, 6(4):553 – 564, aug. 2004. ISSN 1520-9210. doi: 10.1109/TMM.2004.827516.