Video to Audio Conversion for the Visually Impaired

Joseph Redfern

School of Computer Science & Informatics
Cardiff University

Supervisor: Dr Kirill Sidorov
Moderator: Dr George Theodorakopoulos

May 2015

Acknowledgements

Thanks to my supervisor Dr. Kirill Sidorov for his encouragement and support through this project.

Abstract

This final year project report details possible methods of sonfifying video and depth data, with the view to allowing a blind person to navigate a room without any additional assistance.

The report also details and critically evaluates existing solutions to the problem of video to audio conversion, noting their shortcomings and pointing out areas in which they are successful.

By means of an experimental approach, several prototype systems are explained and discussed. Relevant theory has been explained where deemed necessary — including concepts relating to image segmentation, descriptor extraction, shape classification and tone generation.

Contents

List of Figures

Chapter 1

Introduction

World Health Organisation (WHO) figures claim that as as of 2012, there are 285 million people suffering from visual imparements [27], 30 million of whom are blind. The WHO also state that 90% of the visually impaired live in developing countries. Combined with statistics from the guide dogs for the blind association, who claim that the life-time cost of training and keeping a guide dog is around £50,000, a sad picture is painted the majority who are unable to afford such visual aids.

1.1 Aims, Goals and Contributions

This project aims to develop a method of conveying visual information without the user of the system requiring a functional visual system. The study investigates both navigational and semantic modes of operation, and the different techniques associated with each implementation.

The system should be cheaper than current traditional solutions (such as guide-dogs), and be more effective than the more "high-tech" solutions detailed in section 3.3.

The report details a functional system, which is able to be used to effectively navigate around a room and avoid obstacles without the use of eyes. Additionally, the system details the pros and cons of various methods of shape classification and parameterisation, with the view to sonification.

1.2 Intended Audience

The main intended audience for this project is the visually impaired. It is anticipated that the "tech savvy" visually impaired would be interested in trialling the prototypes, both for day-to-day use, and to assist in further development.

Figure 1.1: A guidehorse (source: Wikimedia Commons)

The system is not intended to replace all other forms of visual assistance — rather, it intends to assist those who are unable to afford luxuries such as Guide-dogs or Guide-horses [11]. A low-cost system that is capable of assisting a user in navigating a room and detecting obstacles has the potential to be life changing — even if only a small fraction of a normal, functional visual system can be imitated, 0.01% is infinitely greater than 0.00% [7].

1.3 Project Scope

The project has a fairly broad scope, and includes:

Image Segmentation This deals with the extracting the input image, and extracting an object of interest

Descriptor Extraction This is the extraction of descriptors from the object of interest (obtained from step 1)

Descriptor Sonification Sonification of the output from step 2.

Components/considerations **outside** of the scope of this project are:

Camera Evaluation Detailed evaluation of specific models of depth-sensing cameras is out of the scope of this project.

Robust Testing It is not anticipated that the system described in this report go into immediate production – in-depth safety testing is out-of-scope.

In-depth UX/UI development The UI should be functional, but UI usability is not the primary focus of the project.

1.4 Approach

I took a parallel approach while researching my solution to the problem. Rather than carry out work step-by-step, where problems and delays could have had a serious impact on the project schedule, work was carried out asynchronously to help mitigate such risks.

1.5 Assumptions

Several assumptions have been made during the development of the software. Firstly, it is assumed that the system will be used indoors. The Asus Xtion Pro Live camera [15] chosen for use in the project uses infra-red (IR) laser light to measure distance. In direct sunlight, the IR radiation emitted by the sun over-powers the IR light emitted by the Xtion, resulting in in-accurate readings. This is a limitation of a majority of red, green, blue and depth (RGB-D) cameras using structured light or time-of-flight, although future work could involve the use of a stereoscopic camera setup as a solution to the problem.

Secondly, as the systems described in this report convey information in the form of audio, it is assumed that the user of the system has a functional auditory system. Although there is a percentage of the population that are both deaf AND blind, a majority of the blind have at least some auditory functionality.

Additionally, it is assumed that the potential user of the system would be willing to carry a portable computer (i.e. laptop) and Xtion Camera during it's operation.

Chapter 2

Proposed Solution

An ideal sensory substitution system should convey as much **relevant** information to the user as possible. Bearing in mind the limited bit-rate of the human auditory system (as mentioned in section 3.2), it was decided that the system should have two modes of operation: a navigation mode, and a detail mode, in order to avoid over-whelming the user of the system with data.

It is believed an ideal system will co-operate with the human brain, relying on the user to infer the context of the objects that they can see.

2.1 Navigation Mode

The navigational component of the system is responsible for conveying relevant information about the users surroundings for use when navigating. The use-case for this mode is similar to the use-case for a white cane.

The system should address some of the issues found with the cane — for instance, effective range should be greater than the two paces that a cane provides [8]. Additionally, the system should not, as is the case with a cane, be limited to detecting obstacles on the floor — the field-of-view of the system should be as large as possible.

A specification of such a system is detailed in section 5.4.

2.2 Detail Mode

The "detail mode" of the system should convey detail about a specific object to the user of the system, rather than giving them a general overview of their environment. As many details a possible should be conveyed — the exact amount of information should be determined experimentally.

A more detailed investigation into the plausibility of such a system is detailed in section 5.5.

Chapter 3

Background

3.1 Problem Context

UK Statistics

In the UK alone, around 2,000,000 people live with sight loss — around 360,000 of which are registered with their local authority as being blind or visually impaired, who have severe and irreversible sight loss [4]. Of these 360,000 people, there are only 17,000 white cane-users, and less than 4,800 [3] guide-dog owners. Assuming no overlap between guide dog users and white cane users, this leaves almost 94% without access to the two major forms of assistance available to the blind. It has been estimated that the economic cost of blindness to the UK alone be in the region of £22 billion [22].

A possible factor that could explain the relatively small adoption of Guide Dogs is their cost — as mentioned in section 1, the life-time cost of a guide dog is around £50,000. Although the dog is paid for in its entirety by the Guide Dogs for the Blind association, they themselves are a charity and do not receive government funding. A system whereby the visually impaired could purchase a guide-dog would likely not be effective, as 66% of the registered blind/partially sighted are not in paid employment [10] so would be unlikely to be able to afford the cost.

Developing Countries

The situation in less developed countries than the UK is far worse. According to the Himalayan Cataract Project [35], "blindness is most prevalent in developing countries where malnutrition, inadequate health and education services, poor water quality and a lack of sanitation leads to a high incidence of eye disease". If these countries are so impoverished that they are unable to afford services that are considered basic human rights in the developed world, it is unlikely that they will be able to afford to spend £50,000 per person on guide dogs.

Although white canes are a cheaper, they are not without their downsides. They have limited range — typically a few feet in-front of the user. This makes finding doorways etc a more difficult task than when using a guide-dog.

3.2 The Problem

This project aims to address the lack of a cheap, intuitive way of enhancing the mobility of the visually impaired. It is non-trivial to convey visual information in the form of audio, for a number of reasons.

Compression

Measuring the bit-rate of sensory systems is not an easy task — however, research has been done into the informational capacity of both the human visual system, and human auditory system.

Evidence makes it clear that the human visual system has a higher bit-rate than the human auditory system. Estimates by H. Jacobson [17] suggest the informational capacity for the human ear to be roughly 8×10^3 bits/sec. In a separate paper [18] Jacobson also estimates the informational capacity of the human eye to be around 4.3×10^6 bits/sec — roughly $\times 500$ higher.

Taking this into consideration, it is clear that a working solution to the problem that this project aims to solve will involve compression of the visual information.

Neuroplasticity

Neuroplasticity is a term used to describe the ability of the brain to adapt to change. The brain is able to "re-wire" itself in response to changes to input, environment and emotions — however, this process takes time, and deteriorates with age [28]. Many of the existing solutions detailed in this section of report rely on this neuroplasticity [30].

WHO figures state that of the 39 million blind on planet, 82% of those are aged 50 and above. With this in mind, an inclusive, wide-reaching system should be as intuitive and natural as possible, and should not be too heavily reliant on neuroplasticity.

3.3 Existing Solutions

Several attempts to solve the problem of video to audio conversion have been made in the past.

vOICe

The technique described as "An Experimental System for Auditory Image Representations" [23] sonifies an object/scene by producing a 1:1 mapping from image to audio. This is accomplished by generating a sound, such that a visualisation of the frequency spectrum of the sound produces the input image. This technique has been used by the artist Aphex Twin in the song $\Delta M_i^{-1} = -\alpha \sum_{n=1}^{N} D_i[n] \left[\sum_{j \in C[i]} F_{ji}[n-1] + Fext_i[n^{-1}]\right]$ [34] (more commonly known as [equation]), where viewing the song in a spectrogram produces a picture of the artists face:

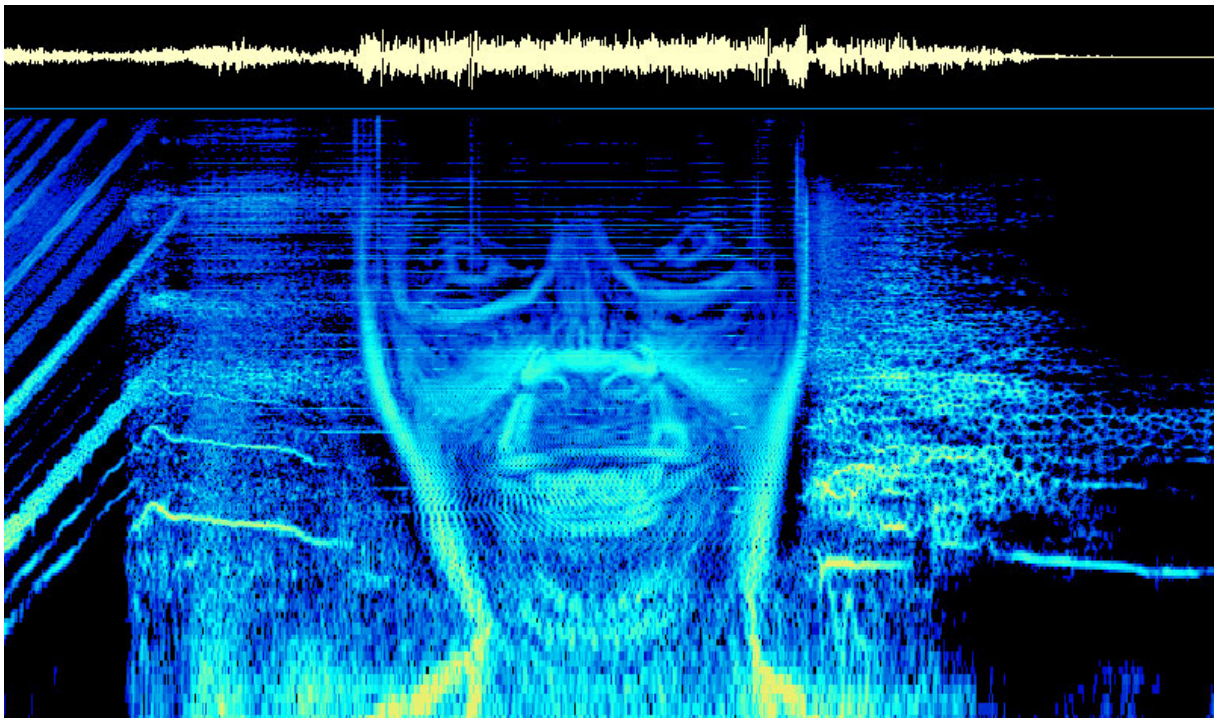


Figure 3.1: Spectrogram of $\Delta M_i^{-1} = -\alpha \sum_{n=1}^{N} D_i[n] \left[\sum_{j \in C[i]} F_{ji}[n-1] + Fext_i[n^{-1}] \right]$ [34]

While this approach can theoretically convey all of the information held within the image, in practice, it is not feasible as a human visual aid. This method would require a human to perform a Fast-Fourier transform (FFT) of the signal and re-construct the image in their head, and assumes that the auditory system has a sufficiently high bit-rate to receive all of the information, making the system unworkable — little compression is performed. Additionally, the data is conveyed to the user in a column-by-column, time-multiplexed fashion, resulting in low temporal resolution.

EyeMusic

EyeMusic [1] is fundamentally the same as vOICe, but with a different choice of sounds (instruments rather than sine waves), and additional image segmentation. It works by clustering the input image into red, green, white, blue and yellow components. Each colour is then assigned an instrument — red is mapped to a reggae organ, green to a "rapmans reed" (a synthesised reeded instrument), white to a choir, blue to brass instruments and yellow to stringed instruments.

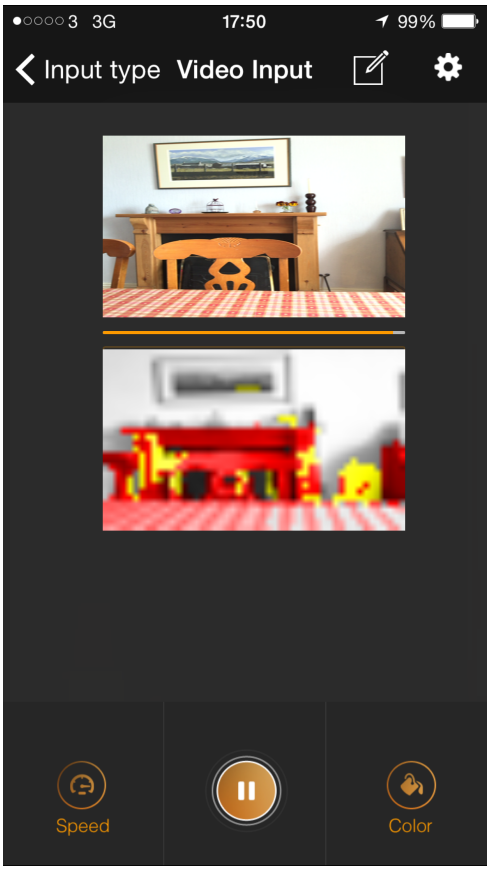


Figure 3.2: EyeMusic iOS Application

The resulting image is then scanned column-by-column, from left to right. A sound is then generated, with instrument varying according to pixel colour, pitch varying according to pixel position on the Y-axis, and volume according to the pixel luminance. The resulting sound is then played back for 50ms (by default), before moving on to the next column.

Virtual Acoustic Space

A paper by Gonzalez-Mora, J.L. et al [12] describes a method involving VAS. VAS works by simulating the sound that a user would hear if a point source at a particular angle and distance from the user was emitting a tone. For each frame, several points are placed — the field of view of the camera is divided into a 17×9 grid, with a point inserted in each division.

The system is not totally dissimilar to echo-location, but uses a simulated response from the objects, rather than relying on an ultrasonic echo.

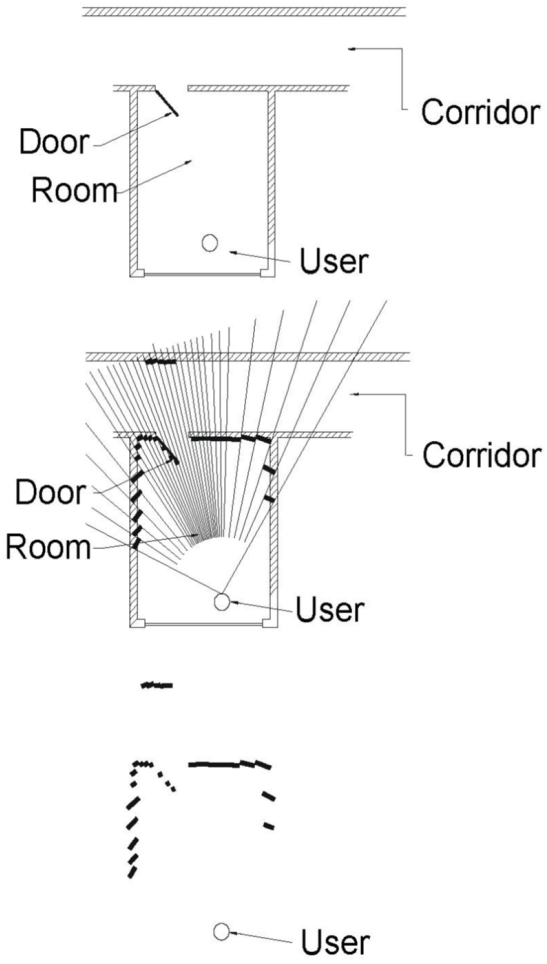


Figure 3.3: Point placement with VAS

The system described uses the Head-related transfer function (HRTF) technique to apply filters to a tone. HRTF works by modelling the effect the human body/head has on incoming audio. For instance, a tone emitted from a source to the left of a user has different properties when received on the left ear to the right ear; Higher frequencies will be attenuated more on the right ear, and there will be a slightly delay between the signal reaching each the right ear-drum compared to the left ear-drum. The HRTF is applied to every point placed by the VAS algorithm on both left and right channels. The modified tones (referred to as pips) are then played back in a random order. By inferring the position of each point based on it's acoustic properties, it has been shown that a trained user can navigate around a room.

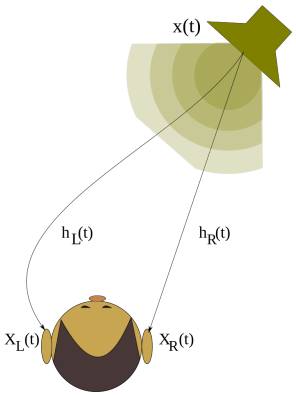


Figure 3.4: Basis of HRTF technique [6]

TheVIBE

"TheVIBE" is a visuo-auditory sensory substitution system [9], which is similar in ways to the method discussed in section 3.3.

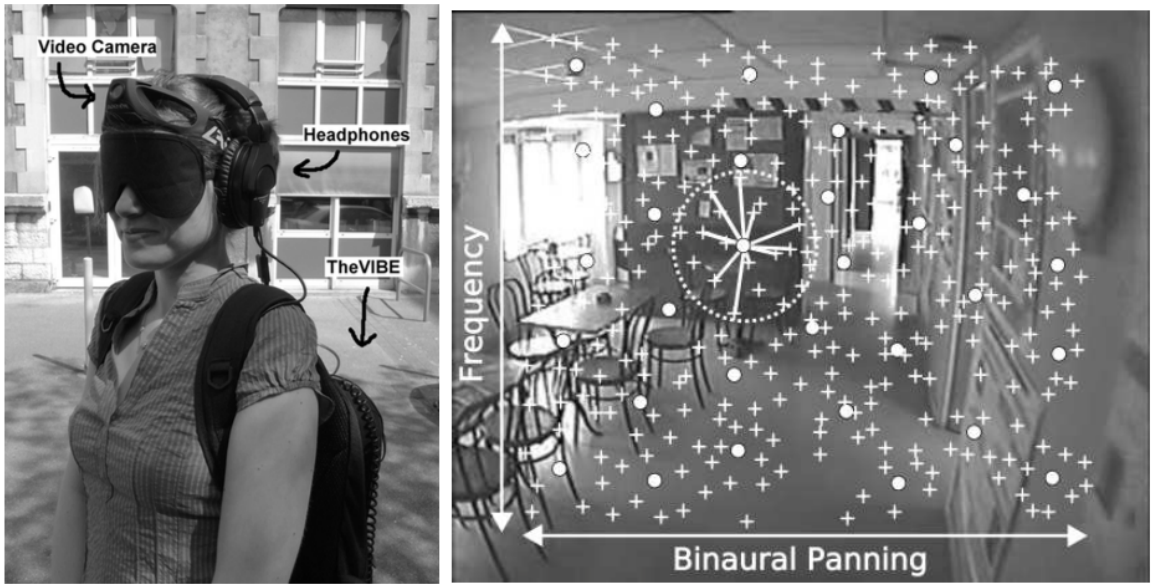


Figure 3.5: Visualisation of point spacing in TheVIBE

This paper describes a technique whereby points are assigned to "receptive fields". Each field has a static position, and is assigned loudness, determined by the Z-axis position of the points within the field. Unlike the method discussed in section 3.3, TheVIBE does not use a full HRTF to convey field position; rather, it uses an inter-aural loudness difference (similar to stereo panning) to describe horizontal position, and tone frequency variation to describe vertical position.

PSVA

Prosthesis Substituting Vision for Audition (PSVA) [5] works by dividing an input image (in this case, from a camera mounted on the users head) into a grid and assigning a

frequency to each pixel, according to the formula:

$$f_n = f_c \times 2^{n/128} \tag{3.1}$$

where f_c is a central frequency, around which pixel frequencies are based, and n is the pixel number.

Additionally, edge detection is performed (through Laplacian of Gaussian) in order to more effectively mimic the processing normally done by the human visual system.

The phase of the sound being generated by each pixel depends on it's horizontal position, with the amplitude of the sound depending on the grey-level intensity of the pixel. The tones are then played back simultaneously (having been generated by an inverse Fourier transformation) and in real-time — the system was prototyped on an Field-programmable grid array (FPGA), presumably due to the limited speed of normal processors at the time of the paper's publication (1993).

With this system, the varying sensitivity of the human eye is accounted for by increasing the size of the pixels at the periphery of the image, resulting in an increased central resolution.

Summary of disadvantages of existing solutions

The existing solutions to the problem of video to audio conversion all suffer from a similar problem — information overload, and training time (due in part to neuroplasticity).

With the exception of PSVA, these solutions are also afflicted with a low temporal resolution, due to the time-multiplexing mode of operation. There is also little prioritisation of information being conveyed to the user, beyond varying the size of the receptive field being performed with PSVA.

Chapter 4

Relevant Theory

4.1 Stereopsis

Stereopsis, translating literally to "Solid Appearance", is the ability to recover depth from one or more image(s). For people with two functioning eyes, this is done sub-consciously by the brain, by observing the differences between the image received by each eye.

Being aware of the depth information in a scene has multiple advantages in the context of this project.

For navigational assistance, it is crucial to be able to know how far away obstacle are, so we are able to warn them that something is blocking their path.

It is also important when conveying the shape or structure of an object to a user. To extract an individual object from an image, we need to know the object boundaries. In a high contrast scenario — for instance, a red object on a yellow background, this is possible using only image data. However, in a lower-contrast situation, for instance a gray object on a black background, this can be more difficult. This report details a method that allows depth information to be combined with image data, demonstrating improvements over image/depth segmentation alone.

As traditional cameras are unable to accurately infer depth, a depth-sensing camera must be used to acquire depth information.

In this section of the report, a basic overview of different ways of digitally acquiring depth information is provided.

Structured Light

The Xtion [15] device used in this project uses a technique involving structured light in order to compute the depth-map of a scene.

The Xtion projects a known pattern of structured IR laser light onto the scene, and using an IR camera, receives the location and shape of each IR point. The camera does not use a lens like those found on normal, visible-light cameras — it has what's known as an astigmatic lens. An astigmatic lens has a focal-length along the X-axis that differs to that along the Y-axis. For instance, if the projected pattern consisted of many circular points, due to astigmatism in the lens, the image read back by the infra-red camera would consist of many elliptical points, varying in eccentricity according to distance between the point and the projector.

Using the change in eccentricity for each point, the device is able to construct a depth-map of the scene in real-time.

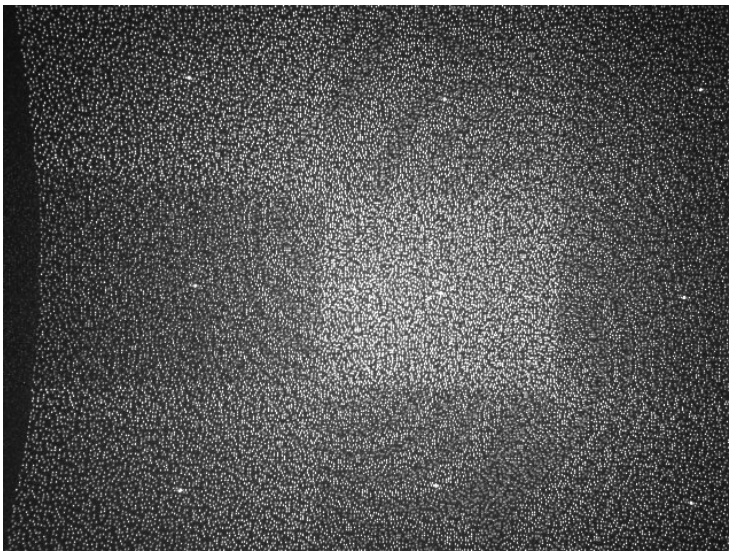


Figure 4.1: IR Pattern from Kinect Device [24]

This sensor was developed by a company called Primesense, and is quite well supported — drivers are available for all major platforms. The Xtion is also supported by OpenNI [26], a framework used to develop software for "Natural Interaction" devices.

Other Techniques

Use of structured light is not the old technique that has been developed to acquire depth-maps - other methods exist, for instance, Time-of-Flight and Stereoscopic systems. The Asus Xtion was chosen over other devices, as it is fairly in-expensive (˜100), and can be powered by USB alone (other devices, such as the Microsoft Kinect, require mains power to operate).

4.2 Image Segmentation

As seen later in this report, this project relies on knowing object boundaries through Image Segmentation. There are many different techniques for segmenting images in Computer Vision — however, there is no "one size fits all" solution, each method has pros and cons.

This section aims to describe some different methods, along with explanations as to why they are/are not relevant to this project.

Histogram Thresholding

Histogram Thresholding is a way of segmenting an image into a binary mask consisting of the background and the foreground. This is achieved by analysing the frequency distribution of image intensity values, and choosing a point on this histogram at which to split the image into two parts.

Figure 4.2: Lena

There are several specific implementations, although they can be generalised into the following form:

$$f : \{0, 1, 2, ..., 255\} \rightarrow [0, 1] \tag{4.1}$$

such that

$$f(0 \cdots T] \rightarrow 0 \tag{4.2}$$

$$f(T \cdots 255) \rightarrow 1 \tag{4.3}$$

where T is the chosen threshold.

Example

Consider this image:

The histogram of this greyscale image is as follows:

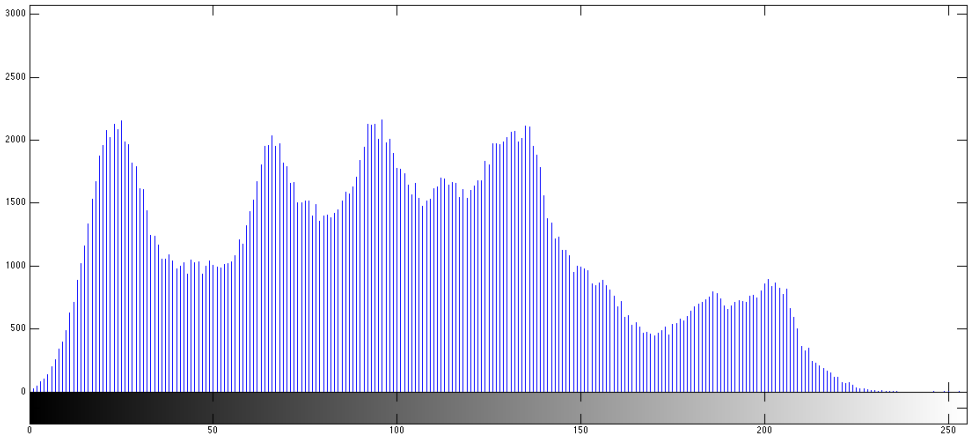


Figure 4.3: Histogram of Lena Image

The peaks on the histogram correspond to the dominant grey tones in the image — dark to light, from left to right.

There are various algorithms to choose the exact point on the histogram at which to threshold, each having their own merit. Automatic, un-assisted thresholding is often considered on of the most difficult tasks in Computer Vision [1], so unless the environment is well-known, alternative methods of segmentation may be more appropriate.

Flood Fill

The flood fill algorithm is used to select an area of an image, based on an initial "Seed Pixel" and a threshold value.

A recursive flood-fill algorithm can be described as follows. Neighbouring pixels to the initial seed pixels are examined — if the difference between the initial seed pixel and the neighbour pixel is less than the given threshold value, then the neighbour pixel is marked as being part of the seed pixel. The algorithm the repeats on all similar neighbouring pixels, always comparing pixels to the **initial** seed pixel.

(a) Initial Seed Point (b) After extraction with tolerance=100

Figure 4.4: Example of flood-fill

This method can work well if the location of the initial seed pixel is known, however, is not so useful if the starting point is unknown.

K-Means

The K-Means algorithm is used to partition a collection of objects into K groups. In the context of Computer Vision, an object is a pixel, and a group is a region.

The algorithm is supplied with K initial colours - ideally (but not necessarily) belonging to the K regions that the image should be segmented into. These colours are known as "Initial Group Centroids".

Each pixel in the image is assigned to the most similar centroid. The value of the centroid is then updated to be set to the mean of it's member pixels.

This process is then repeated, with the value of the centroid being updated with each pass - this can be done a fixed number of times, or until each centroid value converges.

[1] http://www.math.tau.ac.il/ turkel/notes/otsu.pdf

Example

The following is an example of the result of K-means clustering with $K = 3$, with randomly selected Initial Group Centroids.

(a) Input Image (b) K-Means clustering, with $K = 3$

4.3 Elliptic Fourier Descriptors

Theory detailed in this section of the report is used for shape classification, discussed in section 5.5.

Elliptic Fourier Descriptors are the result of performing an Elliptic Fourier Transform (EFT), and can be used to describe the shape of a contour [19].

Before computing the EFD of an image, its contour needs to be extracted. This can be be done by segmenting the image (using a technique described in 4.2), then finding the outermost contour of the segment. This can be done using one of many possible contour tracing techniques, such as Moore Neighbourhood Tracing.

Chain Code

The first step of the Elliptic Fourier Transform is to compute the chain code of the contour.

Chain codes are used to losslessy encode a monochrome image — they describe the connectivity of a list of points, relative to each point.

The best way to explain the idea is through an example.

For instance, consider the following image:

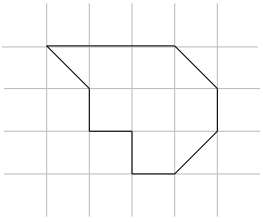


Figure 4.6: Input Image

Each line can be labelled according to the following pattern.

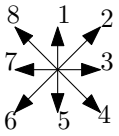


Figure 4.7: Labeling scheme

For instance, an arrow moving right would be labelled 3, an arrow moving down would be labelled 5, etc.

Applying this labelling scheme to the example image results in the following:

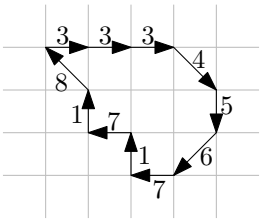


Figure 4.8: Labelled Image

So, the input image encoded as a chain code can be expressed as "33345671718".

Fourier Transform

Once the chain-code of the input has been calculated, the chain code is used to approximate the shape to an ellipse. The accuracy of the approximation will depend on the number of harmonics specified during the transform — the higher the number of co-efficients specified, the greater the accuracy:

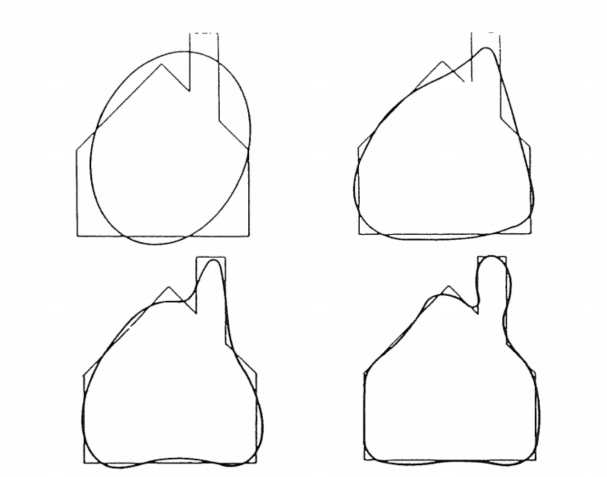


Figure 4.9: Resulting approximations with increasing co-efficents (clockwise), taken from [19]

The specific details of the method used to approximate the chain-code to an ellipse (and the subsequent transforms) can be found in "Elliptic Fourier Features fo a Closed Contour" by Kuhl [19].

4.4 Image Moments

Formally, the moment of an image can be defined as[21]:

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y) \tag{4.4}$$

That is, the sum of pixel intensities, waited according to the x and y co-ordinates of each pixel co-ordinate raised to parameters i and j respectively.

Certain properties of images can be established by using particular i and j values during moment calculation — for instance for a Binary Image, area can be given by: M_{00}, and image centroid can be given by $x, y = M_{10}/M_{00}, M_{01}/M_{00}$. In this way, parallels can be drawn between image moments, and the concept of moment used in mechanics.

Hu Invariant Moments

Hu's Invariant Moments [14] are a set of 7 image moments:

$I_1 = \eta_{20} + \eta_{02}$

$I_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$

$I_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$

$I_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$

$I_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$

$I_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$

$I_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$

where η_{ij} is the scale-invariant moment given by:

$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^{\left(1+\frac{i+j}{2}\right)}}$

and μ_{ij} is the central moment, given by:

$\mu_{ij} = \sum_x \sum_y (x - \bar{x})^i (y - \bar{y})^j f(x, y)$

As the name name suggests, values resulting from calculating $I_{1..7}$ are invariant to scale, rotation and translation — that is, they remain constant no matter their size, their location or angle they are at.

Zernike Moment Descriptors

Zernike Moments are not dissimilar to normal image moments, however are calculated differently — they are based on Complex Zernike Polynomials [20].

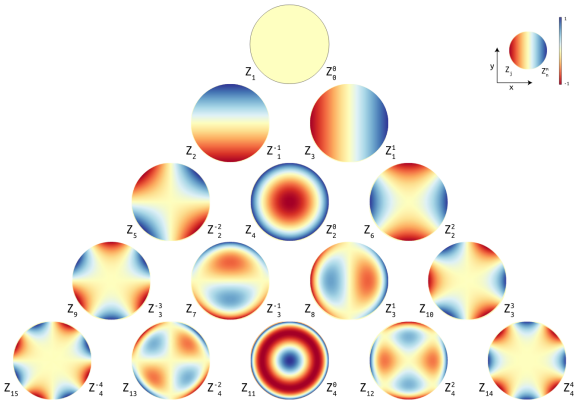


Figure 4.10: First 15 Zernike Polynomials (source: Wikimedia Commons)

Unlike normal image moments, Zernike Moments are orthogonal, and defined on a unit disc — that is, each Zernike moment is statically independent from another, meaning there is little redundancy [2]. This means that using Zernike Moments is an efficient way of parameterizing an image.

Chapter 5

Implementation

5.1 Introduction

This chapter of the report details the experimental approach taken when developing the systems responsible for the two main modes of (Navigation and Detail).

Each mode has differing functionality, but shares some common components — the specifics of these components have been detailed in section 5.3.

5.2 Technical Background

Languages

A majority of the development work done over the course of this project has been in Matlab, although Java and C++ have also been used.

Matlab

It was decided that Matlab was the right tool for the job for several reasons.

Firstly, Matlab has a lot of relevant functionality "baked-in" - for instance, an FFT can be performed on a vector in one line: fft(vector), with no additional libraries required. This is not the case with many other languages. This makes prototyping much quicker, as it is not necessary to totally re-invent the wheel when trying new ideas.

Wrappers for OpenNI are available for Matlab, making interaction with the Xtion camera - the wrapper is open-source, so any modifications can be made quite easily.

Another major factor in choosing Matlab over another language is the amount of support available, both physically from my Project Supervisor, and online from the Matlab community.

The project has been developed using Matlab r2014a, although should be backwards-compatible with most recent releases.

Java

As large parts of Matlab run in the Java Virtual Machine (JVM), it is possible to use Java classes directly from within Matlab. This was taken advantage of in order to use threads, which is something that Matlab does not support out of the box.

Matlab r2014a runs under the Java JVM version 1.7 - as such, any development must be done using the Java 7 JDK or below.

C/C++

C/C++ functions can be exported to the Matlab environment through the use of MEX-files. The OpenNI wrapper used to interact with the Xtion camera was written using C++ and MEX, and required some modifications.

Additionally, a C++ based Video Segmentation [13] tool written by Google and Georgia Tech was trialled, and was also slightly modified.

Data Acquisition

The Asus Xtion Pro Live [15] camera used in this project is compatible with OpenNI, a framework used for developing "Natural Interaction" software.

A module available on the Matlab File Exchange [32] providers a wrapper for OpenNI, allowing the image and depth information to be retrieved from the camera.

Once compiled, the wrapper provides the following functions: `mxNiCreateContext`, `mxNiDeleteContext`, `mxNiPhoto` and `mxNiDepth`.

- `mxNiCreateContext()` initialises the camera and returns a handle to the Xtion device.
- `mxNiPhoto(handles)` returns the current Red, Green and Blue (RGB) from the device referenced at `handles`.
- `mxNiPhoto(handles)` returns the current RGB from the device referenced at `handles`.
- `mxNiDeleteContext(handles)` closes the connection to the camera

Combining these methods, a basic Matlab program to acquire and display the RGB and Depth data from the camera as follows:

```
handles = mxNiCreateContext();
rgb = mxNiPhoto(handles);
rgb = permute(rgb, [3 2 1]);
depth = mxNiDepth(handles);
depth = permute(depth, [2 1]);

subplot(1, 2, 1);
imshow(rgb);

subplot(1, 2, 2);
imshow(depth);
colormap(jet);
```

The calls to `permute()` are necessary as the OpenNI wrapper returns the data in the opposite order to what Matlab expects.

By default, both `mxNiPhoto()` and `mxNiDepth()` return 320×240 images.

The data returned by `mxNiPhoto()` is a 3-channel image containing 8-bit integer intensity values for the Red, Green and Blue channels. The data returned by `mxNiDepth()` is takes the form of a matrix of distance values, measured in millimeters.

5.3 Shared Components

Certain components are shared between both Navigation Mode and Detail Mode.

Magic Wand

For cases where the object of interest is at a known location in the image, a magic-wand style approach was used for object extraction.

Searching the Matlab file exchange, a module able to perform magic wand extraction was found [29]. The module provides a function, `magicwand(im, ylist, xlist, tolerance)` that for a given image, set of x,y co-ordinates, and a tolerance, creates a mask using the flood-fill algorithm described in section 4.2.

Tone Generation

Although Matlab is more than capable of generating and playing back a sine wave, it is not possible to do so while simultaneously performing another task, as it is not possible to write threaded software using normal Matlab syntax. Due to the real-time nature of the system, it was necessary to be able to play back a sine-wave while simultaneously processing video.

In on order to accomplish this, a Java class was developed. This class exposes methods for setting tone frequency, setting tone amplitude and setting pulse duration - internally, a worked tone-generation thread is created. Using this class, it is possible to play back a tone in a non-blocking way.

Internally, the module uses the `javax.sound.sampled` package.

```
BUFFER_SIZE = 2048;
i = 0;
buffer = new byte[BUFFER_SIZE];

while(true){
    samplingInterval = SAMPLE_RATE / getDesiredFrequency();
    angle = (Math.PI*i)/samplingInterval;

    buffer[i%BUFFER_SIZE] = getDesiredAmplitude() * Math.sin(angle);

    if(i%BUFFER_SIZE == 0){
        commitToBuffer();
    }
    i++;
}
```

Support for pulsing the tone was also implemented.

Image Segmentation

Although the primary deliverable of the project was not to invent a new image segmentation algorithm, the method of image segmentation was an important choice.

It was decided that a depth-sensing camera [15] would be used in order to assist with region extraction. Rather than relying solely on RGB data and difference in colour to extract objects from the video footage, depth data would also be considered. This choice was made, as object-segmentation was desired — if a white cup was placed on a white table, traditional colour-based segmentation algorithms may struggle to differentiate between the object and the surface on which the object was sitting.

As majority of image segmentation algorithm implementations consider only colour information for segmentation; it was not possible to use an off-the-shelf Matlab module to complete this task. After reviewing publications on segmentation using both depth and RGB data, a few different approaches were trialled.

Standard Deviation

As an initial experiment, I attempted to highlight "interesting" subsections of the image. This was accomplished by using a sliding window over the RGB image, calculating the standard deviation of the pixels in the window. The resulting image was then thresholded and used on a mask on the original image,

This method was quite successful — the resulting image only contained objects that stood out on the input image.

Figure 5.1: Input Image

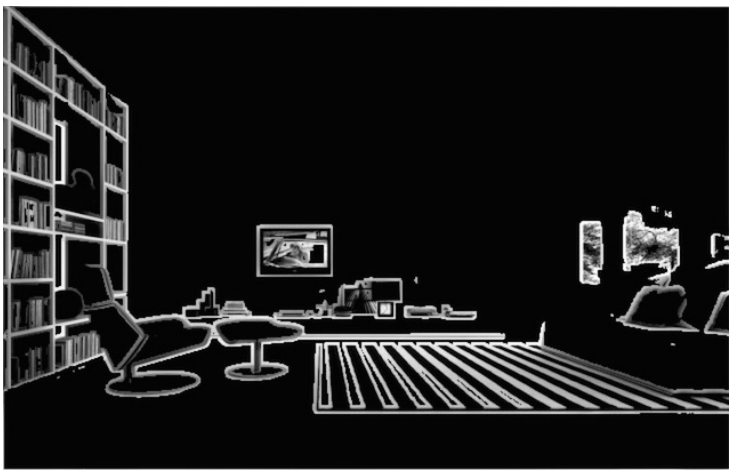


Figure 5.2: Standard Deviation Results

The main issue with this approach was that Depth Information was not used — cases where objects had little RGB contrast would not be picked out.

K-Means with 4 channels

One approach was to use a standard K-Means (see section 4.2) image segmentation algorithm taken from the Matlab File Exchange [33], with an additional channel added, representing depth.

(a) Input RGB Image

(b) Input Depth Map

Figure 5.4: RGB-only K-Means segmentation

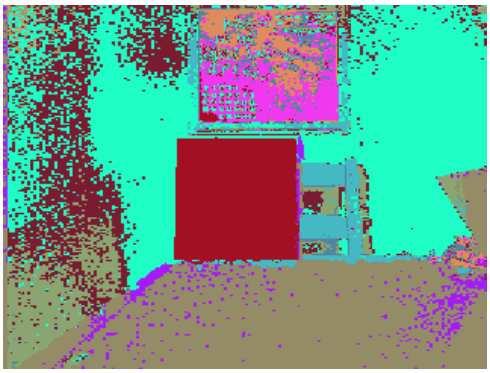


Figure 5.5: RGBD K-means segmentation

These results were fairly good - the addition of the depth channel resulted in a smoother segmentation. However, using raw images from the camera for segmentation

resulted in a fairly noisy image. Applying a gaussian blur to the image ($\varnothing = 5$, $\sigma = 2$) removed some of the noise, at the expense of sharpness in the image.

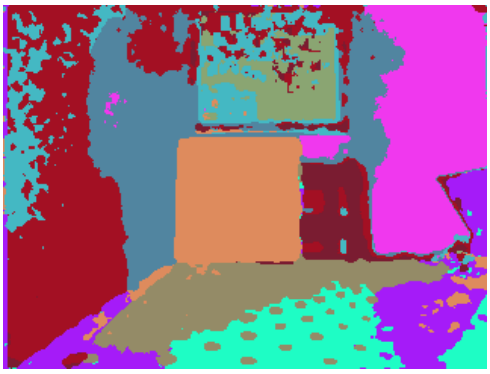


Figure 5.6: Segmentation of blurred RGBD image

Another slight problem with the K-means approach was speed at which it could be performed. While not nearly as slow was some other approaches (such as K-Nearest Neighbour and Graphcuts), performance was limited to the 1-3Hz range.

K-Nearest Neighbour

Channel Swapping

As mentioned, most segmentation algorithms support only RGB data. With this in mind, another approach taken was to remove a colour channel from the RGB image, and swap it with the depth image. This was quite successful — the resulting segmentation was more accurate than either RGB or Depth alone.

Graph Cuts

Attempts were made to use the Graph Cuts algorithm [REF] in order to segment the video and depth data. However, it became apparent that this approach was taking orders of magnitude longer than we could reasonably spend processing each frame; we wanted the system to run as close to real-time as possible.

5.4 Navigation Mode

As mentioned, the main focus of the Navigational of the system is to allow a user to walk around a room and avoid obstacles. Several prototype systems have been developed, each using different techniques to convey different data to the user.

Central Distance

Description

It is possible to read the distance to the center of the image with the following code:

```
handles = mxNiCreateContext();
rgb = mxNiPhoto(handles);
```

```
rgb = permute(rgb, [3 2 1]);
depth = mxNiDepth(handles);
depth = permute(depth, [2 1]);

centralDistance = depth(size(depth, 2)/2, size(depth, 1)/2);
```

Building on this basic idea, a way of conveying this distance was developed. By synthesising a sine-wave with a varying frequency (from low to high) according to distance to the central point, and playing it back in real-time, a user could get a live idea of the distance to the object directly in front of them. As the minimum depth supported by the Xtion is around 550mm, and the maximum is around 4m (4000mm), a direct mapping from distance (in millimeters) to frequency (in Hertz) was used - these frequencies are well within the audible range of the human ear [31].

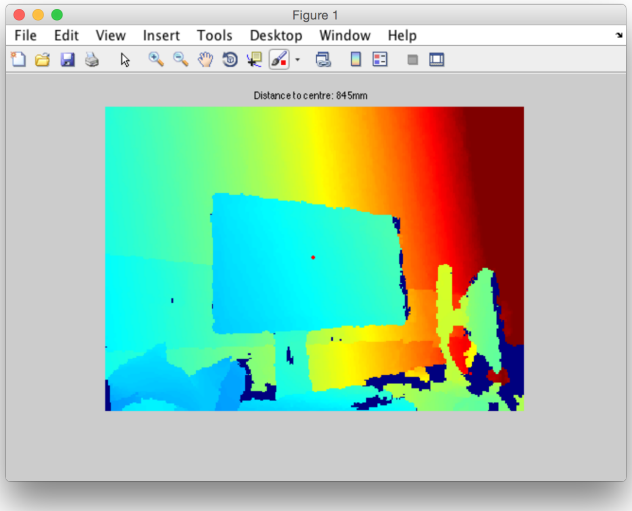


Figure 5.7: Variable-frequency distance system. The red dot marks the center of the depth map

This required the development of the tone-generation Java class detailed in 5.3, and the source-code for this prototype can be found in the file `centralTone.m`.

System Performance

As the algorithm is quite straight-forward, the software implementing it was not excessively CPU intensive. Measuring the performance of the script gave estimates that it performed at roughly 30Hz — this limitation was probably due to the overhead of displaying the image of the depth map using `imshow()`, which was only being done for development purposes.

Effectiveness

The system was tested by walking around a room blindfolded, using only the camera to avoid obstacles. The test was a success — the mapping from frequency to distance was quite intuitive, and there were no "crashes", in terms of both software and the physical sense.

However, I felt that the amount of information being sent to the user of the system could be improved. Given the range of possible frequencies (550-4000), the pitch resolution of the human auditory system (said to be around 1Hz [25]), and the processing rate of the video feed, it is possible to get an approximation of the bitrate of the system:

$$\log_2 (depth_{max} - depth_{min}) \times 30 \approx 350\text{bits/sec} \tag{5.1}$$

Central Distance with Size

Description

Using the 350 bits/sec figure obtained from the entropy calculations in section 5.4 as the base-line to be improved upon, it was decided that additional information could be conveyed.

It was determined experimentally that knowing the size of the object in-front of the user was relevant — with this information, the user knows by how much their will need to alter their trajectory to avoid hitting the object.

It is possible to calculate the physical width and height of the field-of-view using the following formulae:

$$FOV_{width} = 2 \times d \times \tan \frac{58^\circ}{2} \tag{5.2}$$

$$FOV_{height} = 2 \times d \times \tan \frac{45^\circ}{2} \tag{5.3}$$

Where 58°and 45°are the horizontal and vertical Field of View (FOV) of the Xtion camera respectively [16].

This information alone is no more useful to for navigational purposes than the distance to central point - however, by extracting the object at the center of the image, and determining how much of the FOV it occupies, it is possible to calculate the area of the object.

Using the Magic Wand extraction technique described in section 5.3, the central object was extracted as a binary image, where pixels marked 1 indicated that they were part of the object, and pixels marked 0 indicated that they were not.

Figure 5.8: Extracted object as a binary image

An indicator of the real-terms size of the object was calculated using the following formula:

$$\text{Object Area} = \frac{\sum_x \sum_y \text{binaryImage}(x, y)}{\text{ImageArea}} (\text{FOV}_{\text{width}} \times \text{FOV}_{\text{height}}) \tag{5.4}$$

The amplitude of the sine-wave being generated according to the specification in section 5.4 was then varied according to the area calculated using the above formula.

System Performance

The overhead of having to extract the object from the center of the image reduced the frame processing rate to roughly 14Hz, however this slowdown was barely noticable when using the system

Effectiveness

It was proven experimentally that by varying the amplitude of the sound-wave according to the object area, and varying the sound-wave frequency according to the method described in section 5.4, a user could infer both the size of, and the distance to an object.

Multi-point distance

Building on the method developed in section 5.4, a system able to communicate distance to multiple points simultaneously was also designed.

Rather than simply conveying the distance to the central point in the image, a grid of points was used.

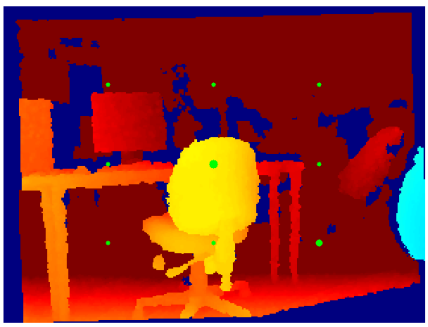


Figure 5.9: Multi-point grid

The left column of points was assigned the left stereo channel, the right column was assigned the right stereo channel, with the center column being assigned both stereo channels.

5.5 Detail Mode

The development of Detail mode was a much more complicated task than the development of Navigation Mode.

The general process can be split into 4 steps:

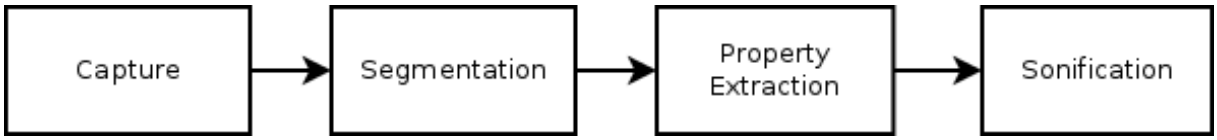


Figure 5.10: Main components of detail mode

The capture step is fairly straightforward — data is read in to Matlab using the Xtion Camera and OpenNI wrapper. Segmentation is the process of extracting relevant shapes from the input image, property extraction is the process of describing the shape in terms of different co-efficients or descriptors, and sonification is the process of conveying those descriptors using audio.

Conveying Shape Properties

Two main ways of conveying details about a given shape were considered.

The first, and perhaps most obvious method was to encode and transmit various shape descriptors directly. For instance, the amplitude of a tone of a particular frequency could convey the area of the shape, with a different frequency being used for number of corners, etc. This method relies on the shape descriptors being intuitively linked to physical properties of the shape.

The second possible method was to transmit details of a shape in terms of similarity to other "basis" shapes.

For instance, consider a method of describing shapes, consisting of 5 parameters. Applying this method to a set of n basis shapes (for instance a circle, square, star and triangle) would yield n sets of 5 parameters. By parametrising an arbitrary shape, and comparing the resulting vector of parameters to the parameter vectors of our basis shapes, it is possible to get a measure of the similarity of our arbitrary shape.

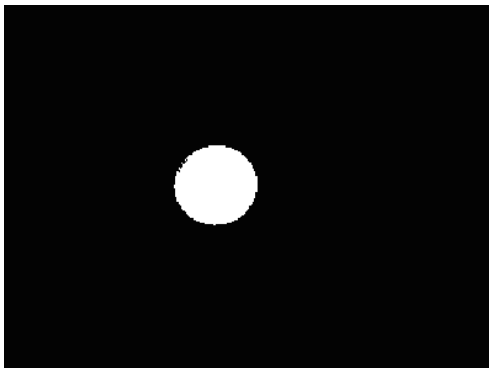
This section of the report details the effectiveness of various methods of conveying different shape properties, using both "direct encoding" and basis shape methods.

Hu Invariant Moments — Basis Shapes

As discussed in section 4.4, Hu's Invariant Moments are seven specific central moments, invariant to rotation, scale and translation.

A method of classifying shapes based on their Hu Invariant Moments was developed. By calculating the invariant moments for a set of basis shapes (in this instance, a circle, square, star and triangle), then calculating the invariant moments for our segmented input shape, and finally measuring the euclidian distance between the moment vector for each basis shape and our segmented shape, it is possible to determine the similarity between the input shape and out basis shapes.

(a) Red, green and blue circular Input Image



(b) Segmented Circular Image (using Magic Wand method)

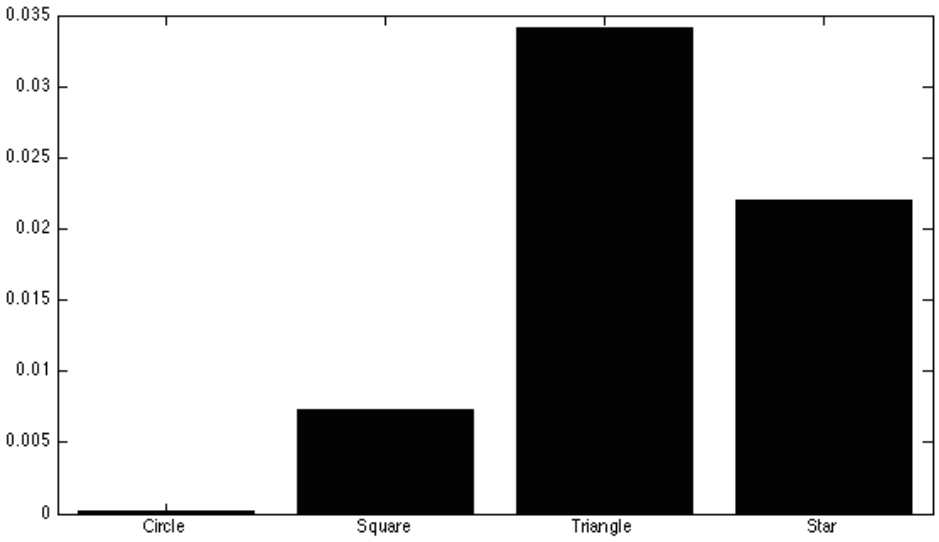
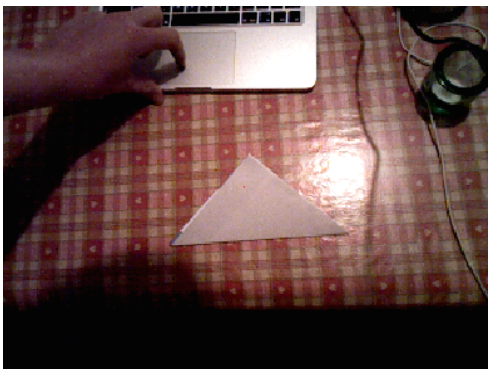


Figure 5.12: Euclidean Distance between basis and input circular shape moments

As you can see, in this example, the system works well — the results correctly state that the shape that is least-different to our input shape is a circle. The experiment was then repeated on some other input shapes:



(a) Red, green and blue triangular input image

(b) Segmented Triangular Image (using Magic Wand method)

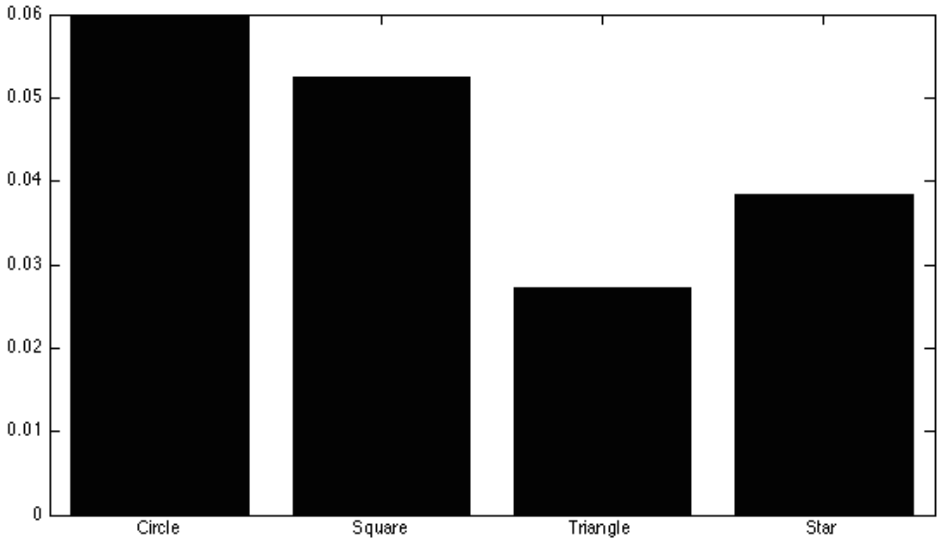


Figure 5.14: Euclidean Distance between basis and input triangular shape moments

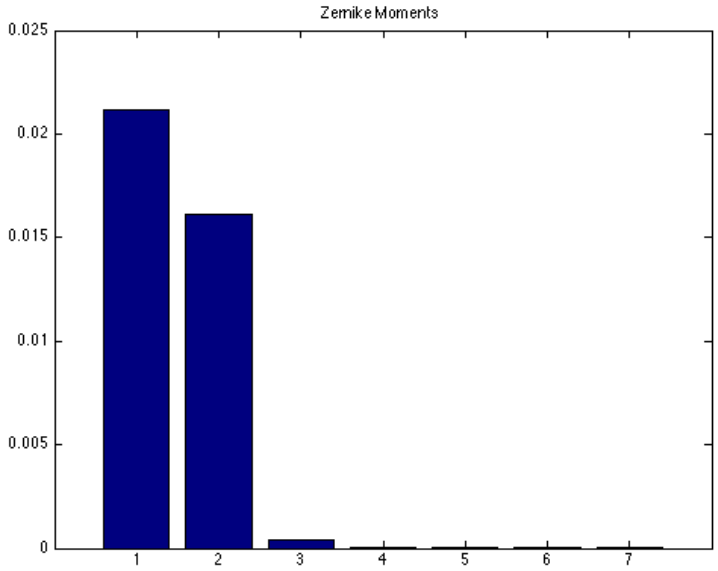
Again, for a triangular shape, the system works well.

There were some cases where in-correct shapes were reported — however, the system was fundamentally a success. Future work relating to this technique will include an evaluation of the best choice of basis shapes.

Zernike Moments — Direct Encoding

By assigning a harmonic to each of the Zernike moments, and varying the amplitude of each harmonic according to the moment value, a tone was generated. I then proceeded to pass various shapes into the algorithm, listening to the tone.

(a) Input Image



(b) Zernike Moments for given input

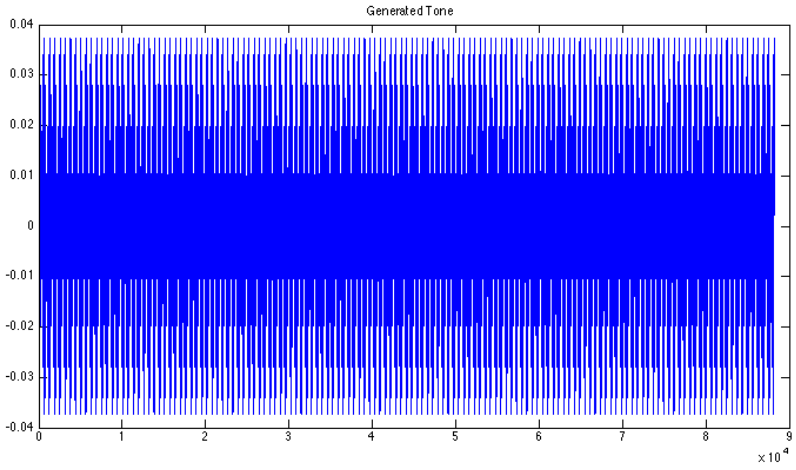


Figure 5.16: Resulting Tone

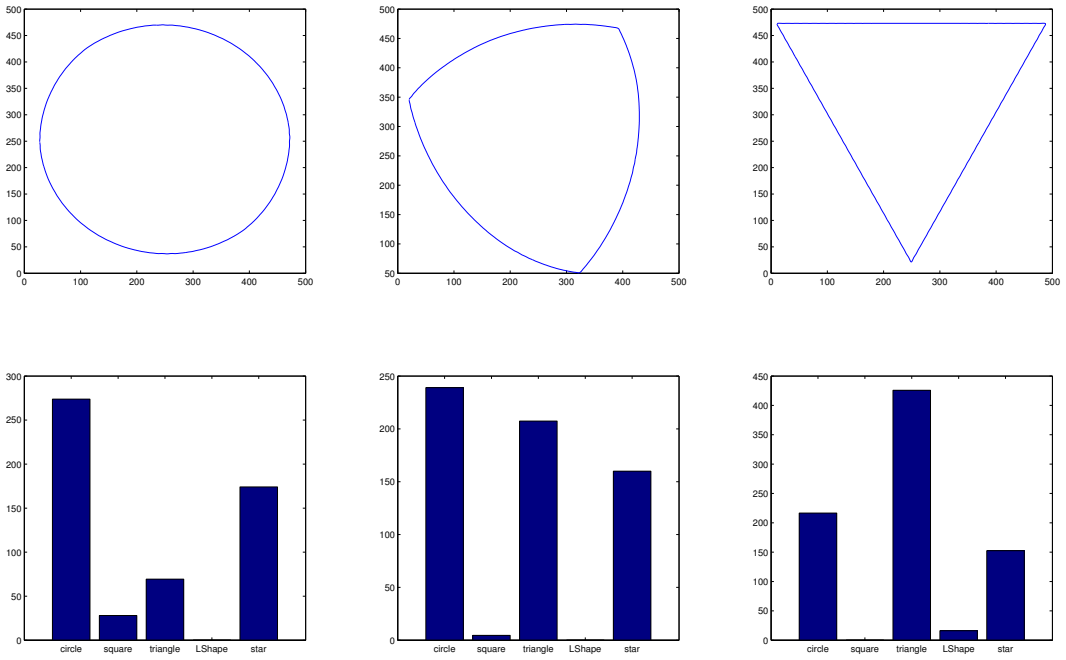
Although the tone varied with each shape, it was not possible to identify individual shapes using this system — it was unclear how a change in the amplitude of a particular harmonic corresponded to changes in shape.

Elliptic Fourier Transform - Basis Shapes

Another approach was to use the Elliptic Fourier Transform (as described in section 4.3) in order to describe the input shape.

Rather than using the approach used with Hu Invariant Moments of computing the Euclidean distance between the basis shapes and the input shape, an alternative approach was experimented with.

It was shown that the shape resulting from interpolation between two input shapes was natural - i.e. the intermediate shape between a circle and triangle was partially circular, and partially triangular:



(a) 100% circle, 0%triangle, 100 harmonics (b) 50% circle, 50%triangle, 100 harmonics (c) 0% circle, 100%triangle, 100 harmonics

Figure 5.17: Interpolation between circle and triangle

The graphs in the above figures show a measure of similarity (based on Euclidean distance) of the input shape to the basis shapes. You can see from subfigure 5.17b that the intermediate shape is correctly measured as pre-dominantly consisting of a mix of triangle and circle.

Exploiting this fact allows us to use the EFT descriptors from the basis shapes within a system of linear equations - i.e.:

$$\text{input} = a \cdot \text{descriptor}_{\text{circle}} + b \cdot \text{descriptor}_{\text{triangle}} + c \cdot \text{descriptor}_{\text{square}} + d \cdot \text{descriptor}_{\text{star}} \quad (5.5)$$

where a, b, c and d are the proportions of circle, triangle, square and star shapes respectively.

This can be solved in Matlab using the `mldivide` method, or the \ operator. Due to a lack of time, this experiment remained unconcluded, however is something that could be addressed in future work.

Chapter 6

Future Work

Although a solution to the problem posted has been developed, there is a large scope for future work based on this report. I have chosen to break this down into 2 key areas.

6.1 Navigation Mode

The proposed multi-point system works in theory, but in practice additional work is required to realise the algorithm. It is not currently possible to emit audio from a single stereo channel, due to limitations in the Java class developed for the prototype. It should be quite trivial to add this functionality, but due to a lack of time, this was not possible.

6.2 Detail Mode

As detail mode was seen as being the most difficult problem solve, it seems natural that it is the problem with the largest scope for future work.

Basis Shapes Selection

In order to continue the work on the basis-shapes proposition (of describing an unknown shape in terms of other, known shapes), research should be done into the optimal choice of basis shapes.

The basis shapes used sections 5.5 and were fairly arbitrary choices — the ideal basis shapes should be experimentally determined.

Additionally, for the sake of completeness, the basis-shape method should be trialled using Zernike Moments as descriptors.

Basis Shape Sonification

Due to time constraints, a limited amount of sonification was done during the basis shapes experiments. Future work in this department will include choosing the optimal choice of sounds — be it sine waves of different frequencies, different instruments, or a single instrument with different notes for each shape.

Chapter 7

Conclusions

To conclude, I believe that this project has, overall, been a success.

Existing solutions to the problem of video to audio conversion for the blind have been described and critically evaluated.

A functional system allowing a blind person to navigate a room has been developed, which addresses some of the concerns raised with existing solutions to the problem.

The feasibility of a system allowing a more detailed view of shapes has been investigated, with initial prototypes for shape identification using various methods having been developed and evaluated. Although ideally more work in this area would have been done, given the size of the problem and my (current) lack of domain-specific knowledge, I believe that a virtually unlimited amount of time could be devoted to this task if such an amount of time were available to me.

Chapter 8

Reflection on Learning

I believe my reflection on learning can be surmised with two quotes — the first being a law coined by Douglas Hofstader, that: "It always takes longer than you expect, even when you take into account Hofstadter's Law", the second having been said by H. L. Mencken: "For every complex problem there is an answer that is clear, simple, and wrong.".

I have enjoyed working on the project immensely, having never undertaken any *serious* research task in the past. With that in mind, if I were to repeat the task, I would have done some things differently.

Firstly, I would have modified the scope of the project to be slightly more realistic, due to a combination of **not** having previously been aware of Hofstaders Law, and due to me now being more aware of the (current) limits to my knowledge. Before starting the project, I think a part of me felt that I would be able to quickly read some articles on the Matlab website and OpenCV Wiki, and instantly be aware of every nuance of Computer Vision that would be relevant to the project. Were I more aware of my own limitations at the time, given the amount of time available to complete the task, I would have reduced the scope of the project.

Secondly, I would have managed and prioritised my time more effectively. I had agree to take part in several events over the course of the Final Year Project, which took away valuable research and programming time. That is not to say that I regret not locking myself in my room for 12 solid weeks — but in the future, I intend to schedule and plan my time more effectively, and avoid leaving things to be quite so last minute.

List of Acronyms

WHO World Health Organisation

IR infra-red

RGB-D red, green, blue and depth

RGB Red, Green and Blue

FFT Fast-Fourier transform

VAS Virtual Acoustic Space

HRTF Head-related transfer function

JVM Java Virtual Machine

FOV Field of View

EFT Elliptic Fourier Transform

PSVA Prosthesis Substituting Vision for Audition

FPGA Field-programmable grid array

Bibliography

[1] Sami Abboud et al. "EyeMusic: Introducing a visual colorful experience for the blind using auditory sensory substitution". In: *Restorative neurology and neuroscience* 32.2 (2014), pp. 247–257.

[2] Gholamreza Amayeh et al. "Accurate and efficient computation of high order zernike moments". In: *Advances in visual computing*. Springer, 2005, pp. 462–469.

[3] Guide Dogs for the Blind Association. 2014. URL: `http://www.guidedogs.org.uk/aboutus/guide-dogs-organisation/facts` (visited on 04/25/2015).

[4] Royal National Institute of Blind People. 2014. URL: `https://help.rnib.org.uk/help/newly-diagnosed-registration/registering-sight-loss/statistics` (visited on 04/25/2015).

[5] C. Capelle et al. "Real time experimental visual prosthesis using sensory substitution of vision by audition". In: *Engineering in Medicine and Biology Society, 1994. Engineering Advances: New Opportunities for Biomedical Engineers. Proceedings of the 16th Annual International Conference of the IEEE*. 1994, 255–256 vol.1. DOI: `10.1109/IEMBS.1994.412057`.

[6] Wikimedia Commons. *Hrtf diagram.png*. 2005. URL: `https://commons.wikimedia.org/wiki/File:Hrtf_diagram.png` (visited on 04/24/2015).

[7] Richard Dawkins. *The Blind Watchmaker: Why The Evidence Of Evolution Reveals A Universe Without Design Author: Richard Dawkins, Publisher*. WW Norton & Company, 1996.

[8] Jason Dowling. "Mobility Enhancement using Simulated Artificial Human Vision". PhD thesis. Queensland University of Technology, 2007.

[9] Barthélémy Durette et al. "Visuo-auditory sensory substitution for mobility assistance: testing TheVIBE". In: *Workshop on Computer Vision Applications for the Visually Impaired*. 2008.

[10] *Facts and Figures about issues around sight loss*. 2006. URL: `https://goo.gl/YH5vBr` (visited on 04/25/2015).

[11] The Guidehorse Foundation. 2005. URL: `http://www.guidehorse.com/` (visited on 04/21/2015).

[12] J.L. Gonzalez-Mora et al. "Seeing the world by hearing: Virtual Acoustic Space (VAS) a new space perception system for blind people." In: *Information and Communication Technologies, 2006. ICTTA '06. 2nd.* Vol. 1. 2006, pp. 837–842. DOI: 10.1109/ICTTA.2006.1684482.

[13] Georgia Institute of Technology Google Research. *The Video Segmentation Project.* 2014. URL: https://github.com/videosegmentation/video_segment (visited on 04/24/2015).

[14] Ming-Kuei Hu. "Visual pattern recognition by moment invariants". In: *Information Theory, IRE Transactions on* 8.2 (1962), pp. 179–187.

[15] ASUSTek Computer Inc. *Xtion PRO LIVE.* URL: http://www.asus.com/uk/Multimedia/Xtion_PRO_LIVE/ (visited on 01/29/2015).

[16] ASUSTek Computer Inc. *Xtion PRO LIVE.* URL: http://www.asus.com/uk/Multimedia/Xtion_PRO_LIVE/specifications (visited on 04/26/2015).

[17] Homer Jacobson. "The informational capacity of the human ear". In: *Science* 112.2901 (1950), pp. 143–144.

[18] Homer Jacobson. "The informational capacity of the human eye". In: *Science* 113.2933 (1951), pp. 292–293.

[19] Frank P Kuhl and Charles R Giardina. "Elliptic Fourier features of a closed contour". In: *Computer graphics and image processing* 18.3 (1982), pp. 236–258.

[20] Simon X Liao and Miroslaw Pawlak. "Image analysis with Zernike moment descriptors". In: *1997 IEEE Canadian Conference on Electrical and Computer Engineering (IEEE, 1997).* Vol. 2. 1997, pp. 700–703.

[21] Simon Xinmeng Liao and Miroslaw Pawlak. "On image analysis by moments". In: *Pattern analysis and machine intelligence, IEEE Transactions on* 18.3 (1996), pp. 254–266.

[22] Access Economics Pty Limited. *The economic impact of partial sight and blindness in the UK adult population.* 2009. URL: https://www.rnib.org.uk/sites/default/files/FSUK_Summary_1.pdf.

[23] Peter B. L. Meijer. "An Experimental System for Auditory Image Representations". In: *IEEE Transactions on Biomedical Engineering* 2.2 (1992), pp. 112–121.

[24] Anand Muglikar. *Accessing IR Video Stream from PrimeSense 3D Sensor using OpenNI and OpenCV.* 2013. URL: http://stomatobot.com/primesense-3dsensor-ir-stream/ (visited on 04/28/2015).

[25] R. Nave. *Sensitivity of Human Ear.* URL: http://hyperphysics.phy-astr.gsu.edu/hbase/sound/earsens.html (visited on 04/26/2015).

[26] Inc Occipital. *OpenNI Home-page.* 2014. URL: http://structure.io/openni (visited on 05/05/2015).

[27] World Health Organization. *Visual imparement and blindness.* 2014. URL: http://www.who.int/mediacentre/factsheets/fs282/en/ (visited on 01/28/2015).

[28] Denise C Park and Gérard N Bischof. "The aging mind: neuroplasticity in response to cognitive training". In: *Dialogues in clinical neuroscience* 15.1 (2013), p. 109.

[29] Son Lam Phung. *Simulating Photoshop's magic wand tool*. 2004. URL: http://www.mathworks.com/matlabcentral/fileexchange/4698-simulating-photoshops-magic-wand-tool/content/magicwand.m (visited on 04/27/2015).

[30] Ella Striem-Amit. "Neurplasticity in the blind and sensory substitution for vision". In: (2013).

[31] Peter Howell Stuart Rosen. *Signals and Systems for Speech and Hearing*. Emerald, 2011, p. 163.

[32] Camillo Taylor. *Matlab Wrapper for OpenNI 2.2*. 2013. URL: http://uk.mathworks.com/matlabcentral/fileexchange/42127-matlab-wrapper-for-openni-2-2 (visited on 04/24/2015).

[33] 'Thorsten'. *Colour Image Segmentation using K-Means*. 2015. URL: http://uk.mathworks.com/matlabcentral/answers/58952-colour-image-segmentation-using-k-means#answer_71396 (visited on 04/06/2015).

[34] Aphex Twin. $\Delta M_i^{-1} = -\alpha \sum_{n=1}^{N} D_i[n] \left[\sum_{j \in C[i]} F_{ji}[n-1] + Fext_i[n^{-1}]\right]$. 2005.

[35] *World Blindness Overview*. 2015. URL: http://www.cureblindness.org/world-blindness/ (visited on 04/25/2015).