80158: Human Computer Interaction
Design and evalution of a user interface for a motion analysing application

Date: Tue 0:00, 31.05.16

Prof. Camurri, PhD Radosław Niewiadomski

Rabbia Asghar, BEng, Ernest Skrzypczyk, BSc, Jessica Villalobos, BSc

Contents

List of Figures

Chapter 1

Design of the user interface

Application and user analysis

One of the most important aspects for the user interface (UI) design has to be dealt with before the design process itself. A proper analysis of the application, ergo task and environment, as well as a user analysis are required to be performed first. The presented case deals with a UI for medical personal, a therapist that deals with body posture problems. The target user is assumed to have little to none background in programming. The used environment was not precisely specified, however it is assumed a medical praxis or hospital. Several other assumptions have been made and certain goals set as described in following sections, however some aspects had to be dropped in the process of the design, mostly because of software limitations.

Application scenario

The main task of used software is to detect and help with body posture problems of patients. In order to perform the analysis of motion pictures of patients an EyesWeb patch is being used. The software can provide a variety of information extracted from provided videos. There are several parameters the user can concentrate upon.

Task analysis

For a proper medical analysis the data extracted from video material needs to be precise. This is not always possible, since the original video material might be of overall low quality, especially lightning should be of influence. In certain scenarios the patients might not be able to go to a medical praxis. Therefore basic pre-processing options should be available for the user in order to allow for modifications of the video material. Even with high quality video material a precise analysis might be only possible, if filtration parameters are exact for a given case. Also another important aspect is the ability to compare the video material before and after processing, so that the user can confirm the algorithm is working correctly in the case of the specific patient and also be able to catch relevant details from both videos.

The human body is complex in its entire structure. Therefore a movement limitation or incorrect performance might need analysis from different perspectives. This would require specification of many options, some might be more relevant than others for performed tests. Many options require attention. In order to offload mental work from the user, a linear approach in the design of the UI that also categorizes options in a preferably intuitive manner should be taken.

Certain injuries, deformations or disabilities are more common than others. It can be assumed with a high probability that same type of movement limitations would use the same or at least similar settings for the software. This implies that presets of settings might be very valuable for the user with respect to avoiding repetitive task and save time, providing more opportunity to deal with the actual problem.

The structure of the performed task is assumed to stay the same or at least follow a certain principle. There are no constraints on software performance other than to provide a processed video that runs smooth enough to actually recognize movement, but this aspect might be out of reach for the user design itself and should not be considered further.

The task requires a highly skilled user in movement therapy with a basic understanding of computer interaction and video processing. The target user might also use other software at the same time, therefore a capability to work with multiple applications in a standard desktop environment should be provided.

No conflicts should occur during the task, which is assumed to be performed as often as needed, however with the possibility of being conducted periodically. Further task related aspects are being considered in following sections.

Environment analysis

No specific information about the environment of the performed application have been given. However it is assumed that the analysis takes place in a medical praxis or a hospital, where rooms are properly equipped and there is no noise for the actual task, in case of movement analysis for example other people moving in front of camera.

Depending on the room where exercises are performed by subjects, time of day and the inner light illumination, the lightning can influence the readability and ability to recognize defects or aspects of interest in the computed data of the user. Also the movement analysis could be performed all day long, tiring the user, therefore the interface should have the capability of minimal impact on the vision sense should be provided.

User analysis

According to given specifications the target user is a therapist, highly skilled in movement of the human body. The target user has however no programming skills and is required to have basic experience in using a computer. Therefore the design of the user interface should be kept simple, focusing on the task and not subjecting the user to insignificant data, which could lead to information overload and faster tiring. Models, widgets or principles that a therapist is familiar with should be preferred.

No further specifications in respect to the user have been made. The provided information are basically restrictions from the point of view of user interface design. However this should not limit in providing capabilities to the more advanced users that are very well acquainted with the software and technological background.

Since the target user is expected to be educated in the medical field of therapy, further information can be extrapolated. It is expected the user is highly literate and motivated, with a positive attitude, capable of communicating very well with patients. The interface should provide precise and technically relevant to the task information, while being neutral to the user conducting the movement analysis.

Because of a rather straightforward approach no personas or scenarios have been developed for the task of designing a user interface.

EyesWeb

The EyesWeb project is a research platform, that is open source and deals with design and development of real-time multimodal systems and interfaces. The designing capabilities allow to develop a user interface for the required task. A software patch for the main frame has been provided with the capabilities of motion analysis that can be parametrized.

Sketch

The designing process has been roughly conducted according to the user centered design, which is based on an iterative approach. The first step has been conducted in the previous section *"Application and user analysis"*. The second step consists of a simple sketch of the UI. The figures (1.1) and (1.2) illustrate the first iteration sketch.

The sketch has been done according with the user, task and environment analyses. The structure of the UI consists of a main window, that is always present, and sub-windows, very much like modal slide-out windows, with settings and options that can be called from the main window. The main state is marked with the number 1 and for the sliding sub-windows the state of the UI is marked with the number 2.

Main

Since the video material is the main information source, it has been granted the most space, similar to the processed data represented in a graph and coordinate values, which holds valuable information from the EyesWeb patch for the target user, the therapist. Ideally also standard video control would have been included, but this idea was dropped, because of EyesWeb's limitations.

The design should be intuitive for the user and since the task is dealing with measurements and analysis of body motions, the approach of placement of widgets in the main window resembles a heart monitor or an oscilloscope, both of which are measurements devices that the target user has had a high probability of dealing with. Also most people are right-handed[1], this design takes advantage of that fact. Thus, the main visual feedback is located on the top left of the main screen, while button and sliders, referring to switches and dials from the hardware example, are on the right and bottom of the window.

The top right space of the main window is dedicated to first parametrization of the task, namely the choice of video input. Since a comparison between patients or different examinations of the same patient can be conducted often, a *Recent files* drop-down menu list with files that have been recently access is to be implemented. Next to it is the *Load* button responsible for loading an user selected video material. The *Recent files* should hold the base name of the currently loaded file in visible section, when the drop-down menu is not activated. Also there should be a *"..."* fielf on the top of the active drop-down list.

To allow the user to quickly compare the processed and unprocessed videos, the main window also holds several video specific quick access options.

The next option from the top is the *Split videos* button. As noted in the sketch, the main video section on the top left is supposed to be composed of one or two video screens side by side. This button switches between these two arrangement patterns and also disables the *Video mixing* part.

Further down are the *Un/Dock* buttons *Input*, *Output* and *Graph*, each corresponding to the undocking and docking sub-windows, ergo detaching and attaching the them from the main UI, so that they can be used in parallel to other applications. *Input* refers to the video input, ergo the unprocessed material, and *Output* to the processed one and *Graph* the actual graph displaying analysed data.

Another important tool is the slider with buttons *Video mixing*. Users of audio or video processing hardware or software, should recognized this widget combination. The slider provides the user the capability to mix input and output video streams, so that a precise comparison for each video can be made. Since the slider requires the user to point the mouse cursor to it, then click, hold, move and release the mouse button on the appropriate position, it takes a long time and effort. For convenience and to save time buttons *A* and *B* have been placed on the ends of the slider, intuitively suggesting minimum and maximum values. This concept of *A* and *B* streams mixing is very popular, however labels additionally explain that *A* stands for *Input* and *B* for *Output*, ergo the slider at most to the left outputs only the input video, and conversely at most to the right the processed output video. Anywhere between the video are mixed according to the set ratio.

[1] Wikipedia – Handedness

These three simple methods give the user the capability to switch quickly between different comparison patterns of the video material. The position of each of those widgets corresponds to the likelihood of usage from the top to down, with the top *Split videos* being of most likely used as preferred method by majority of users. Again, the target user is required to have only basic understanding of video processing, therefore it is not of the highest priority, however expert users will appreciate this easily available option.

At the bottom of the main screen each of the sub-windows has a toggle button: *Presets*, *Effects & options* and *Body*. Those names have been chosen with consideration to the target user, who is not required to poses expert computer or video processing skills and expertise. The names are short, informative and serve their purpose. A name *Centroids and coordinates* would be a poor alternate to *Body*, even though it is correct, it might be confusing and even intimidating to the user.

Presets

Since usually a therapy does take multiple sessions, sometimes spread over years, many examinations of patients take place. A therapist might have hundreds of different patients. To allow the target user to concentrate on the given task, a sub-window with *Presets* has been developed. This modified state of the UI, where the sub-window appears when the button is active is marked on the sketch with the number 2.

The presets as already indicated in section *"Application and user analysis"* are holding the settings for the application, so that change between different sets of options and settings can be performed quickly. The user should also have the ability to save the current settings set and load a previously saved one. Since the application is designed for a specific application and target user as well as environment, it the optimal approach would include implementing gathered therapist and video processing experts experience into a predefined set of presets in an easily accessible list. Those should correspond to the most probably and most common settings and patient cases.

Effects & options

The *Effects & options* section holds the rest of video and general processing specific settings.

Starting with background settings including *Threshold* and *Filter order* for video processing. The widgets for those parameters consists of sliders for quick adjustment and text boxes for precise input, which might be necessary especially when the video quality is poor.

Next section holds *Display options* consisting of radio buttons for quick selection and unambiguous feedback of the active processing algorithm. For active *Ghost* display option, next section holds the parameter *Ghost number*. Since there are only a few options, the radio button widget has been used again. If *Ghost* is not active, the radio buttons for *Ghost number* get disabled.

Color subregions in the next section provide the user the capability to change the color of detected subregions using combo boxes. Again the widgets get disabled if the appropriate processing option is not selected.

Another section deals with *Features extraction* using radio buttons once more and the last section *Smoothness* holds a centroid for possible parameter values.

This sub-window deals mostly parameters that need to be set only once per video, therefore they do not hold high priority.

Body

The *Body* screen would be the main sub-window of attention to the target user. It holds *Centroids* and *Coordinates* buttons on top setting the mode of the sub-window.

The main part of this sub-window is the skeleton on the left, which corresponds to a widely used wooden doll in arts and medicine, therefore the target user should be familiar with. Also this use of the image and arrangement of buttons on the joints and other points of interest makes this part of the interface intuitive.

This section works differently depending on the selected mode by buttons *Centroids* and *Coordinates*. Buttons should offer visual feedback when active.

The two list boxes on the right correspond to the skeleton and are also in two different modes selected by main buttons. Left box lists all possible and still not selected *Body parts*, while the list box on the right shows *Active* and selected body parts. Between the list boxes are three buttons <, > and *Clear* dealing with activating and deactivating a selected option resulting in moving it between the two list boxes or clearing all active body parts.

The last section *Colors* allows the user to toggle the visibility of processing features for *Centroids* and *Lines* as well as select the colors from a predefined list consisting of 16 high contrast colors. Also for custom colors except of putting the RGB value directly into the combo box, there is a *Color picker* button next to the combo boxes. When active a modal window with the well know color picker interface should be shown. The UI should accept both decimal and hex values.

General comments and guidelines

The presented sketch has been done focusing on the user centered design. Since the target user can have a limited experience with task related settings, all widgets and relevant parts of the user interface should hold tooltips. Furthermore visual feedback is quite relevant and should be provided as well, especially where ambiguity in the UI could occur to the user. Additionally color themes should be implemented to allow the user to work in different or dynamic lightning conditions, where especially for longer working sessions the dark high contrast themes would tire the eyes less, due to less illumination and therefore energy being absorbed by the eyes. Lastly color coding for sections and groups of related objects, as well as space-to-priority ratio of widgets, especially buttons, should be upheld. Familiar interface segments can enhance the user experience of the UI, therefore a metaphor to a wooden doll would be intuitive to a therapist, who have a high probability to encounter and use those in real life. Figure (1.3) represents such a metaphor (1.3a) with a real world object (1.3b) as reference.

After the sketch of the first iteration of the user interface design, small changes, many of aesthetic nature, mostly due to *"Limitations"* have been introduced resulting in the final designed user interface shown in the figures (2.1) to (2.4) in section *"Experiments"*.

Limitations

Several limitations of the software suite EyesWeb resulted in design changes, mostly in restriction to planned to be implemented features. This could have been prevented if the designers would have had a deeper understanding of current capabilities of the EyesWeb Designer. However there are certain aspects that even though included in the sketch, would have required too many resources, especially in form of time and programming effort to be fully implemented, even more so because of the used graphical language. Following limitations have been found specific to the task and influenced the design process as well as the final user interface:

- Main window size should have been minimal, so that the user can with with other software. EyesWeb does not allow currently to implement slide sub-windows, therefore the size of the main window needs to be as big as the biggest sub-window, so that all widgets are displayed. This a major limitation of one of the central aspects of the design approach taken.
- Tabs widgets had to be used instead of sub-windows. The visibility of tabs is being toggled using the buttons at the bottom of the main screen *Presets*, *Effects & options* and *Body*. Because of that approach switching to an inactive tab is possible using one mouse click on the appropriate button, however switching back to an already active tab requires two mouse clicks on the appropriate button: one to switch off the visibility of the desired tab, and another to actually attain focus of the tab. This problem should be solvable within the current capabilities of EyesWeb.

- Undocking and docking of video screens has not been implemented.
- Not all processing options actually influence the processing output. This is a issue with the patch and EyesWeb itself.
- Most probably, because of the way the *Split/Merge Videos* feature has been implemented using 3 video screens, where visibility it toggled in the UI, and all tabs holding those videos, the overall performance of the user interface sometimes can barely meet the required smoothness in the videos displayed. This is however an issue with EyesWeb again. It should not assign resources to not visible video material.
- *Recent files* feature was not implemented.
- No video controls, because of lack of simple access in the main application.
- Presets mechanism have not been implemented properly.
- Color combo boxes with 16 high contrast colors have been reduced to 3 and do not accept custom input values.
- Color picker for *Centroids* and *Lines* was not implemented.
- Visual feedback is only available where it is absolutely necessary, as it is in case of the *Body* sub-window with *Centroids* and *Coordinates* buttons and the skeleton. However the visual feedback for the buttons for specific body parts had to be dismissed, because of the complexity of implementation.
- Tooltips.
- Two list boxes in the *Body* sub-window corresponding to the skeleton.
- Color themes and proper colorization of the interface, ergo color coding, have been skipped or only partially implemented, due to the very high failure rate of the design software and inconsistent results.

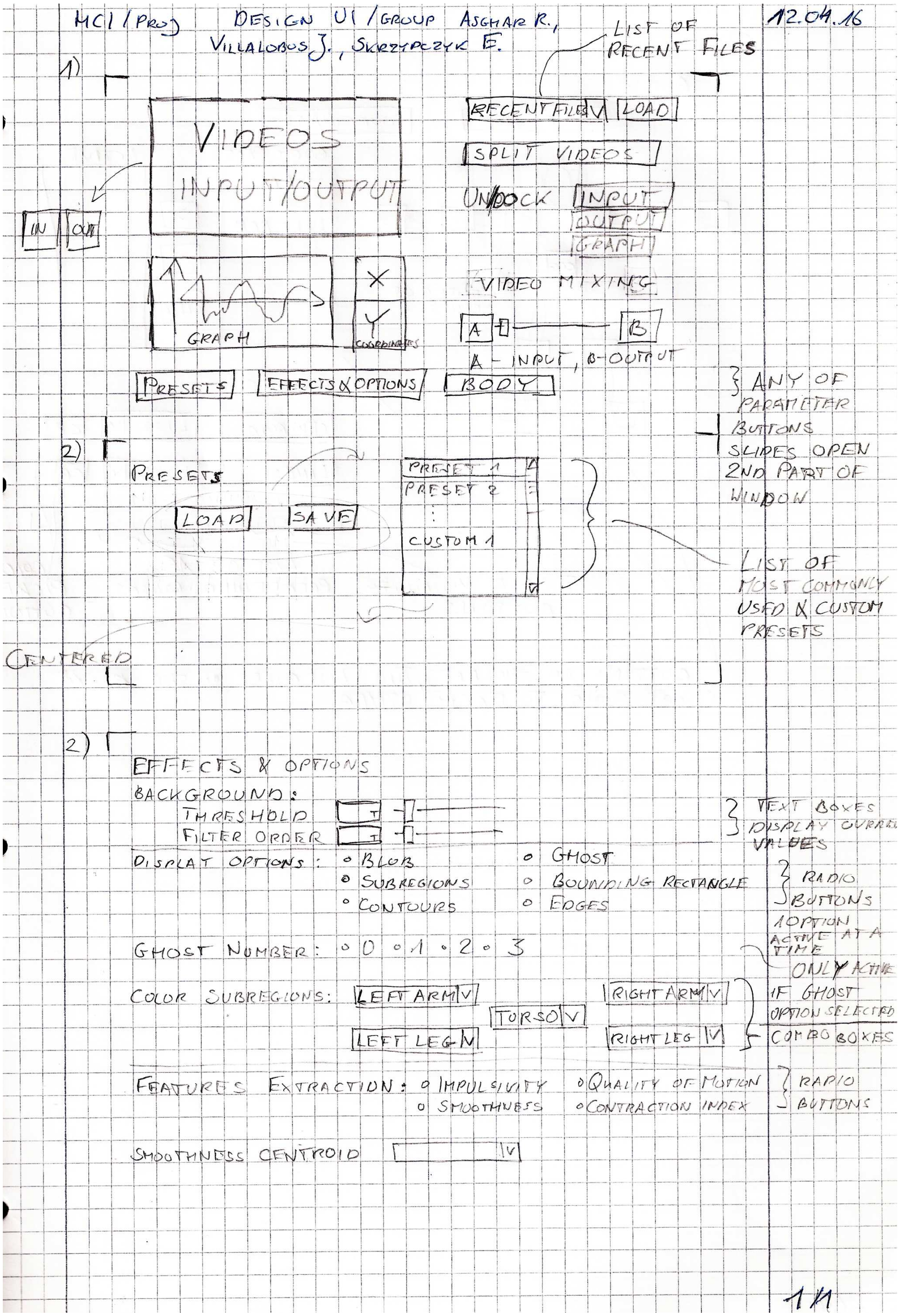


Figure 1.1: Initial sketch of the user interface design – page 1.

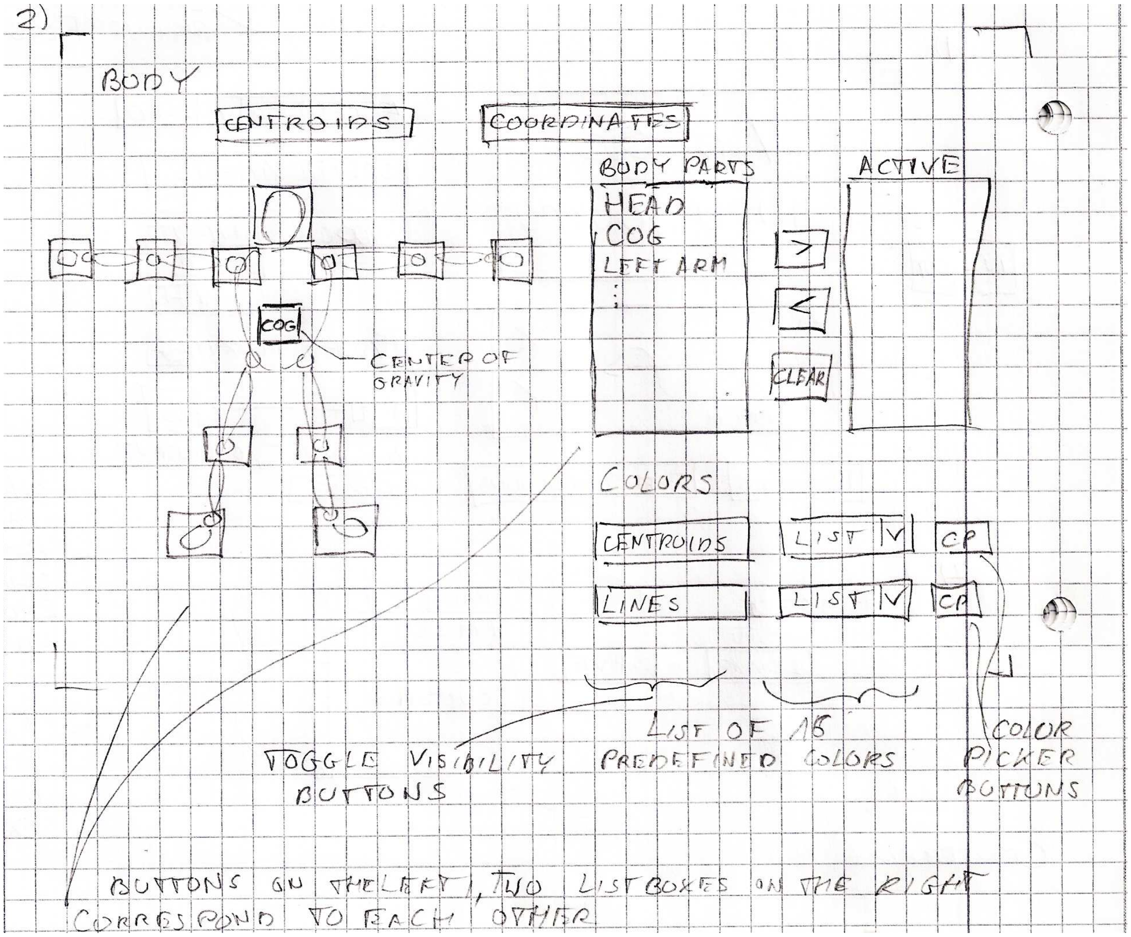
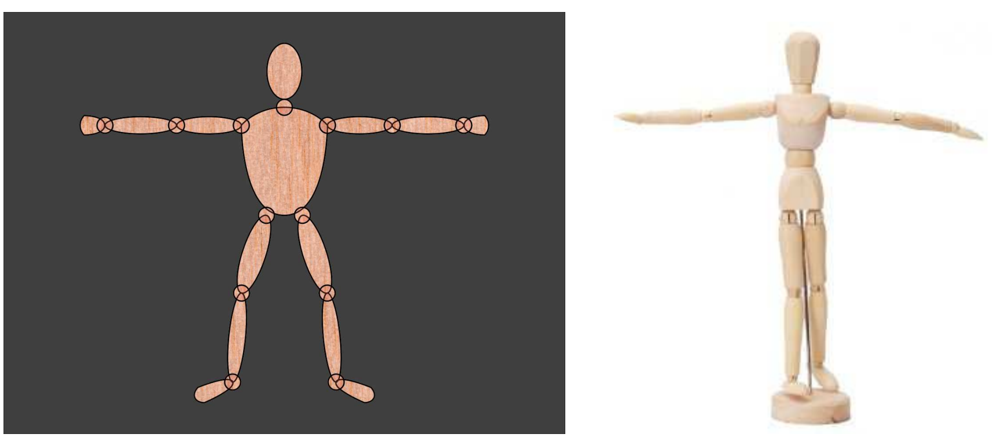


Figure 1.2: Initial sketch of the user interface design – page 2.



(a) A developed graphic for the user interface

(b) A real world object serving as a reference for the metaphor

Figure 1.3: A humanoid wooden doll with movable joints and body parts.

Chapter 2

Evaluation of the user interface

Statistical tools

In order to evaluate and compare user interfaces statistical tools can be used. Two tools were selected for this evaluation enabling a sanity check of the results:

1. Microsoft Excel being part Microsoft's Office package, it is a widely applied spreadsheet software allowing the user to perform different calculations, graphs, pivot tables and limited programming.

2. PSPP is a free software application to analyse sampled data equipped with a graphical and command-line user interfaces.

The *T-test* and the *ANOVA* were the only candidates for the kind of experiments selected as described in section *"Experiments"*.

Experiments

Evaluation and analysis of the UI was performed between two different graphical user interfaces (GUIs).

Figures (2.1), (2.2), (2.3) and (2.4) show screenshots of the developed user interface in accordance with the description of the process from previous sections, most notably *"Sketch"* and *"Limitations"*. Worth pointing out is the vast amount of space wasted in order to accommodate for EyesWeb's limitations as portrayed in figure (2.1). From here on this interface is referred to as *GUI A*.

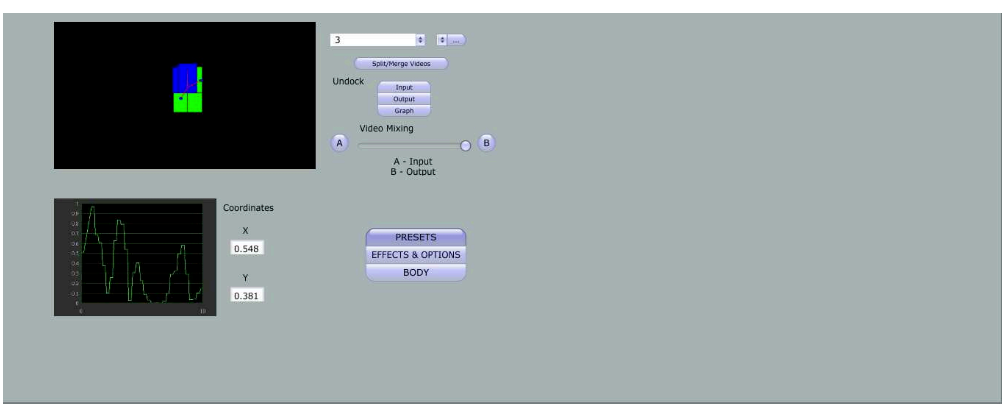


Figure 2.1: Designed graphical user interface – main screen.

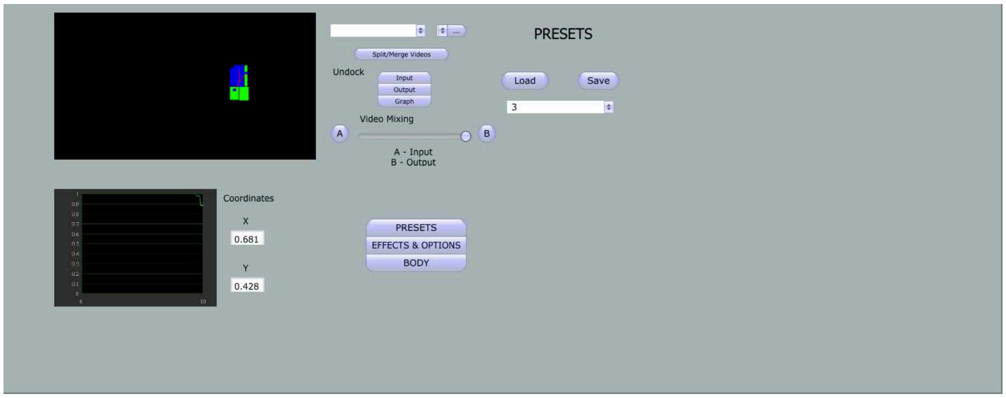


Figure 2.2: Designed graphical user interface – presets.

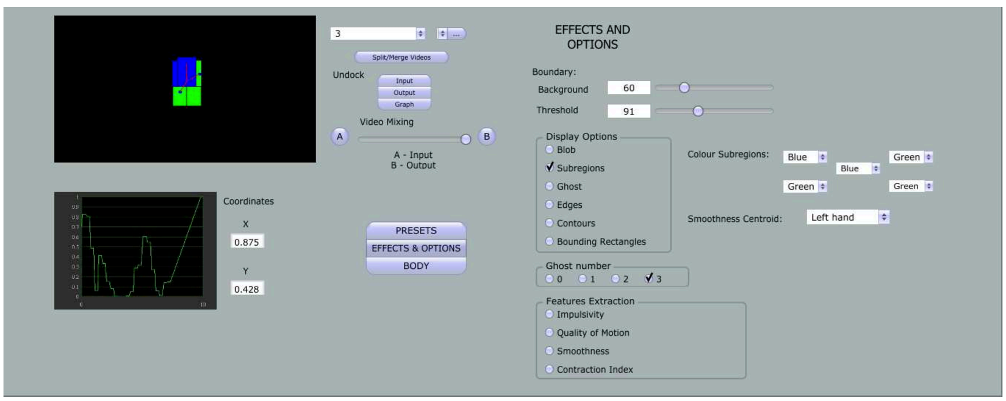


Figure 2.3: Designed graphical user interface – effects.

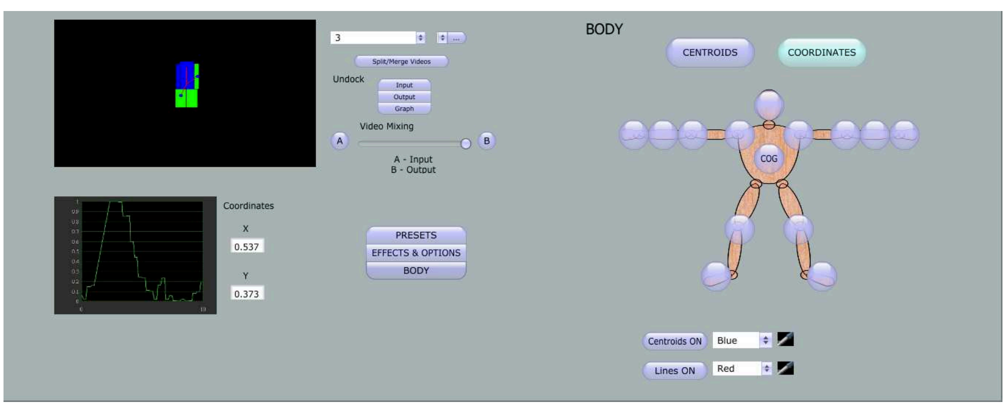


Figure 2.4: Designed graphical user interface – body.

Figures (2.5) and (2.6) show screenshots of another user interface that was used as a reference for comparison to the designed GUI. From here on, this interface is referred to as *GUI B*.

A single task was defined, as described in *"Task definition"*, which was performed by one group of participants using both user interfaces. This technique of conducting experiments in the field of human computer interaction (HCI) is called a *within subjects* test method and is a powerful statistical analysis

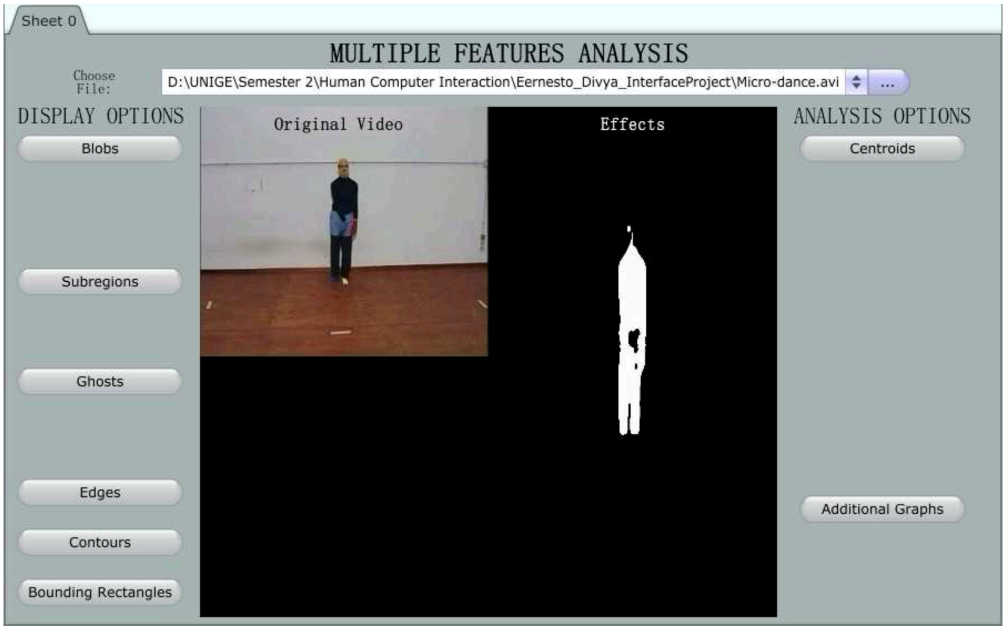


Figure 2.5: Graphical user interface used for comparison – default screen.

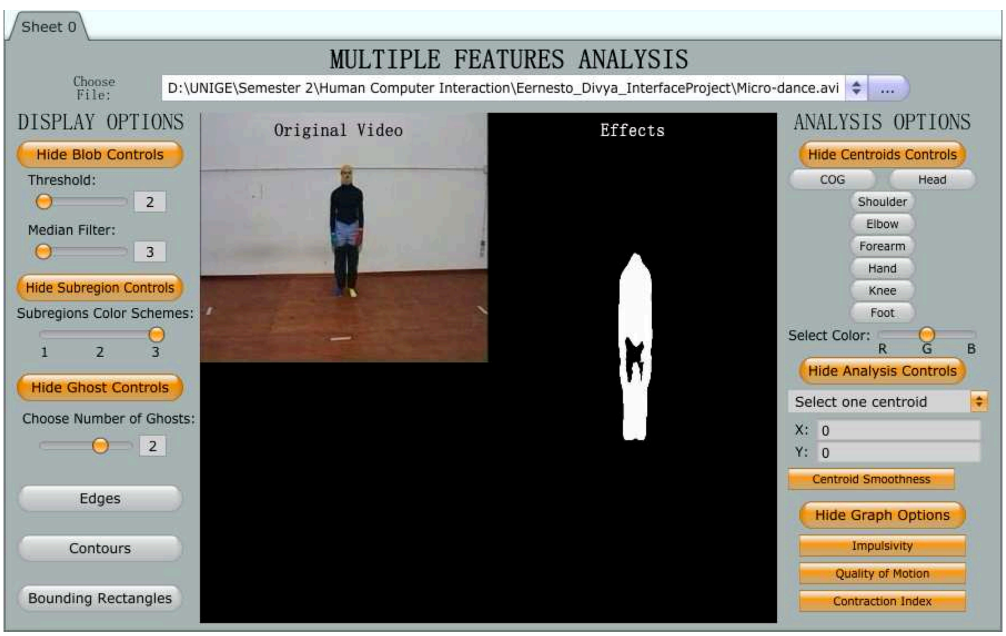


Figure 2.6: Graphical user interface used for comparison – settings opened.

tool. By computing the difference of performance within each user, contribution and personal traits that are unique to that user are cancelled out, limiting individual noise of the measurements and therefore the whole experimental evaluation. On the downside, the *within subjects* technique induces the *carry-over effect*, ergo the users are inclined to perform differently in subsequent trials due to experience gathered in previous trials.

For data analysis the two-sided, within subjects T-test was chosen to be performed, given that this test allows the same group of people to perform the task in both GUIs. The test group consisting of 15 people was allowed to familiarize itself with both user interfaces and gain certain experience using them. It is assumed that this greatly reduced the *carry-over effect*.

Task definition

A task that could be performed using both GUIs was defined and composed of the following set of actions:

1. Select the video file to be processed.
2. Set appropriate filter settings:
   - background selected at 25% (60),
   - threshold selected at 33% (90).
3. Enable centroids and select the following parts to be displayed:
   - centre of gravity (COG),
   - right hand,
   - head,
   - left knee.
4. Select subregions as processing option.
5. Change the color for centroids to blue.

The listed actions were not defined according to the linear workflow provided by the design of both GUIs, especially *GUI A*. This non-linear requirement in order of performed actions was implemented deliberately to test the *ease of navigation* through the user interface.

Measurements

Initially three different measurements were planned to be conducted for analysis and evaluation of the provided graphical user interfaces *GUI A* and *GUI B*:

1. Time to perform a well defined task with non-linear set of acions (*"Task definition"*).
2. Errors committed by user.
3. Ask the user to rate the GUIs according to *ease of navigation*.

The test subjects were allowed to familiarize themselves with both GUIs and gain experience in their operation before the measurements. Users were then given the task. Time was measured between the moment when the GUI started running and the user signalled readiness until all subtasks were performed. Since this measurement was solely dependent on time taken to perform the task irregardless of user's subjective perception, it was considered as an objective measurement.

During the first set of conducted experiments, it was observed that users made very few to no errors. This was most likely due to simplicity of the task with respect of prior familiarization with the GUIs and gained experience. Therefore, with the assumption of no relevant additional information, the measurement for errors committed by users during the task performance was thus removed.

At the end of the test, users were asked to rate the two GUIs based on the Likert scale described in table 2.1 according to the ease of navigation. This comprised of subjective measurement.

Table 2.1: Likert Scale

1	2	3	4	5
Strongly agree	Agree	Neither	Disagree	Strongly disagree

Figure (2.7) presents the data collected from both measurements. Part (2.7a) enlists time taken by each user to complete the assigned task, while part (2.7c) shows means and standard error of the means for each GUI. Part (2.7b) enlists Likert scale ratings given by each user for both GUIs, while part (2.7d) shows corresponding means and standard error of the means.

Standard error of the mean (SEM) is the standard deviation of the sample mean's estimate of a population mean. It is computed using the formula in equation (2.1).

$$SEM = \frac{standard\ deviation}{\sqrt{number\ of\ samples}} \tag{2.1}$$

Analysis and evaluation of measured data

Both the user interfaces show the processed video at all times. This helps the user in observing the result and impact of changes in settings without the need to switch windows as it is common with certain desktop environments. However, *GUI A* offers different tabs to change parameters, while *GUI B* provides all options on the same page resulting in an information overload. According to Miller's Law[1], the short term memory allows to store 7 ± 2 chunks of information. Too much information at the same time can result in decreasing the performance of user.

It is assumed that the different approach in the designs of *GUI A* and *GUI B* will make a significant difference in time taken and ease of navigation in performing the task.

Objective measurement

First, the hypothesis is presented that there is a statistically significant difference in time required for performing the task between the two different GUIs. This hypothesis aims to reject the null hypothesis stating that both interfaces offer the same mean time to perform the task.

Figure (2.7a) enlists time taken to perform the task using *GUI A* and *GUI B* in seconds. Two-sided within subjects T-test is performed for analysis using both presented statistical tools. T-test is conducted on the basis of two assumptions:

1. The data set is normally distributed.
2. The variances of two measurement data sets are equal.

In order to perform the analysis using Excel, the means and standard errors of the means (2.7c) are computed from the individual data sets to generate the bar chart (2.8). Standard error is indicated on the bar charts in form of vertical error bars (2.8). It can be observed that error bars of the means from both data sets are not mutually exclusive indicating that the mean time taken for performing the task by users is similar for the two GUIs. This can be further verified by increasing the number of participants for future evaluation tests.

A paired T-test illustrated in (2.9) showed no significant difference between the *GUI A* and the *GUI B* condition (two-tailed $t = 0.016$, $df = 14$, $p = 0.99$). It took similar time to accomplish the task using *GUI A* (mean: $77.1s$) and using *GUI B* (mean: $76.9s$).

[1] Wikipedia – Miller's law

GUI A	GUI B
41	71
101	74
64	75
52	95
98	83
72	125
46	73
80	87
132	77
110	81
65	91
75	41
90	50
60	73
70	58

(a)

GUI A	GUI B
1	2
4	2
2	3
3	1
4	5
1	4
2	4
5	1
5	2
1	3
2	5
2	4
1	2
3	1
3	1

(b)

GUI A	**GUI B**
77.0666667	76.9333333
6.27904215	4.91841588

(c)

GUI A	**GUI B**
2.6	2.66666667
0.35023801	0.36106837

(d)

Figure 2.7: Measured data for evaluated graphical interface designs.

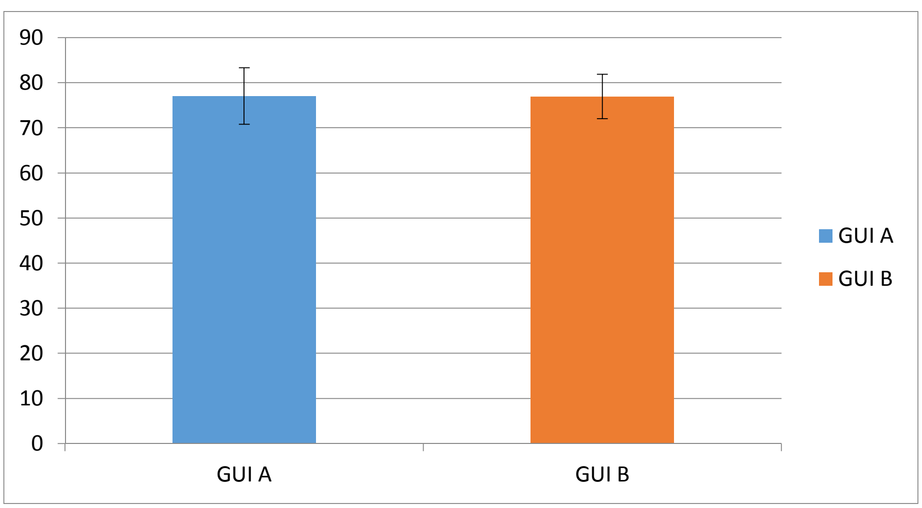


Figure 2.8: Time analysis for both graphical user interfaces with average values and standard error bars.

	GUI A	*GUI B*
Mean	77.0666667	76.93333
Variance	633.638095	388.7809
Observations	15	
Pearson Correlation	-0.05093547	
Hypothesized Mean Difference	0	
df	14	
t Stat	0.0157648	
P(T<=t) one-tail	0.49382225	
t Critical one-tail	1.76131014	
P(T<=t) two-tail	0.9876445	
t Critical two-tail	2.14478669	

Figure 2.9: T-test for time analysis using Microsoft Excel.

A detailed T-test was performed using PSPP. The assumption based on normal distribution of the data sets was verified by generating histograms and by performing the one-sample Kolmogorov-Smirnov (K-S) test. It can be observed on the histograms shown in figure (2.10) that the data is normally distributed. This is further verified from the result of sample K-S test (2.11). The asymptotic significance for *GUI A* and *GUI B* is 0.954 and 0.705 respectively. Since these values are higher that 0.05, the assumption that the data sets are normally distributed is satisfied.

Performing paired samples t-Test (2.12) in PSPP yielded the same results as in Microsoft Excel, confirming accuracy and integrity of the calculations.

The analysis and evaluation of the measured data show that the hypothesis of a statistically significant difference in time required for performing the task between the two different GUIs is *incorrect*.

Subjective measurement

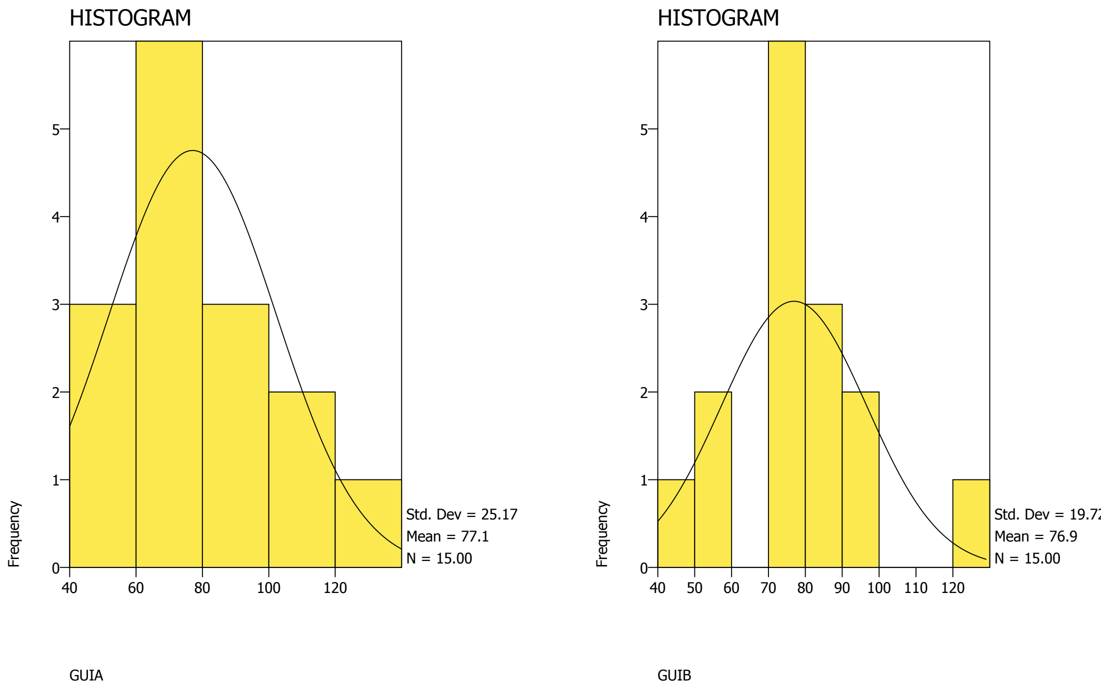
In this case, the hypothesis is that there exists a statistically significant difference in the ease of navigation between the two GUIs. This hypothesis aims to reject the null hypothesis stating that the users find both GUIs user friendly at the same level with respect to the rating of the ease of navigation.

Figure (2.7b) enlists the Likert scale rating of *GUI A* and *GUI B* by users. Similar to the objective measurement, a two-sided within subjects T-test is performed for analysis using Excel and PSPP. T-test is conducted on the basis of the same assumptions as before.

Analysis using Microsoft Excel again starts with computation of means and standard errors of the means (2.7d) from the individual data sets in order to generate the bar chart (2.13). Standard error is indicated on the bar charts in form of vertical error bars (2.13). Again, it can be observed that error bars of the means of both data sets are not mutually exclusive indicating that the mean Likert scale ratings by users is similar for the two GUIs. Again it is suggested to increase the number of participants in order to confirm this result.

A paired T-test (2.14) showed no significant difference between the *GUI A* and the *GUI B* condition (two-tailed $t = -0.11$, $df = 14$, $p = 0.913$). The participants rated both GUIs equally well with *GUI A* mean rating equal to 2.6 and *GUI B* to 2.67.

Using PSPP a detailed T-test was performed. The assumption based on normal distribution of the data



(a) (b)

Figure 2.10: Histograms of data sets for time analysis with superimposed normal distribution curve.

		GUIA	*GUIB*
N		15	15
Normal Parameters	*Mean*	77.07	76.93
	Std. Deviation	25.17	19.72
Most Extreme Differences	*Absolute*	.13	.18
	Positive	.13	.11
	Negative	-.08	-.18
Kolmogorov-Smirnov Z		.51	.70
Asymp. Sig. (2-tailed)		.954	.705

Figure 2.11: One sample K-S test for time analysis for both graphical user interfaces.

sets was verified by generating histograms and by performing the one-sample Kolmogorov-Smirnov (K-S) test. It can be observed on the histograms in figure (2.15) that the data is normally distributed. This is further verified from the result of sample K-S test (2.16). The asymptotic significance for *GUI A* and *GUI B* is 0.594 and 0.518 respectively. Since these values are higher that 0.05, the assumption that the data sets are normally distributed is satisfied.

Performing paired samples t-Test (2.17) in PSPP yielded the same results as in Microsoft Excel.

The analysis and evaluation of the measured data show that the hypothesis stating that there is a statistically significant difference in the ease of navigation between the GUIs is *incorrect*.

Paired Sample Statistics

		Mean	*N*	*Std. Deviation*	*S.E. Mean*
Pair 1	GUIA	77.07	15	25.17	6.50
	GUIB	76.93	15	19.72	5.09

Paired Samples Correlations

		N	*Correlation*	*Sig.*
Pair 1	GUIA & GUIB	15	-.05	.857

Paired Samples Test

		Paired Differences							
					95% Confidence Interval of the Difference				
		Mean	*Std. Deviation*	*Std. Error Mean*	*Lower*	*Upper*	*t*	*df*	*Sig. (2-tailed)*
Pair 1	GUIA - GUIB	.13	32.76	8.46	-18.01	18.27	.02	14	.988

Figure 2.12: T-test for time analysis data set using PSPP.

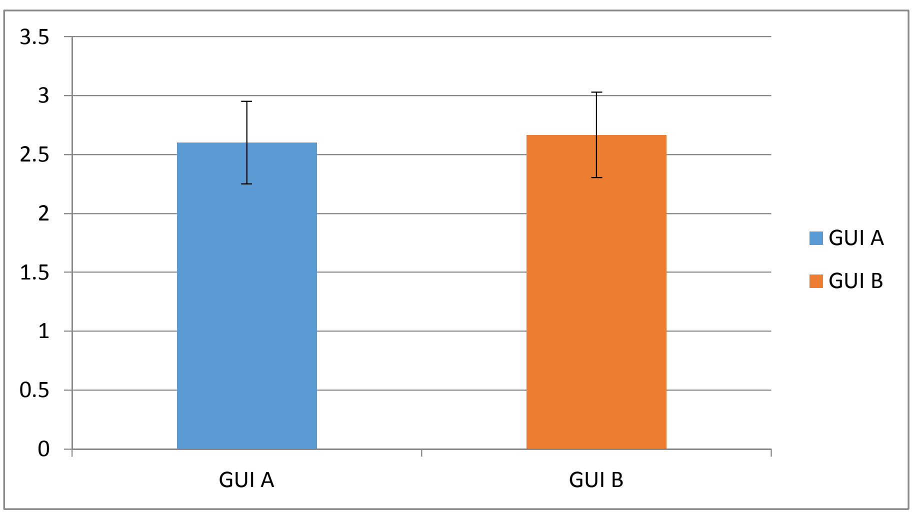
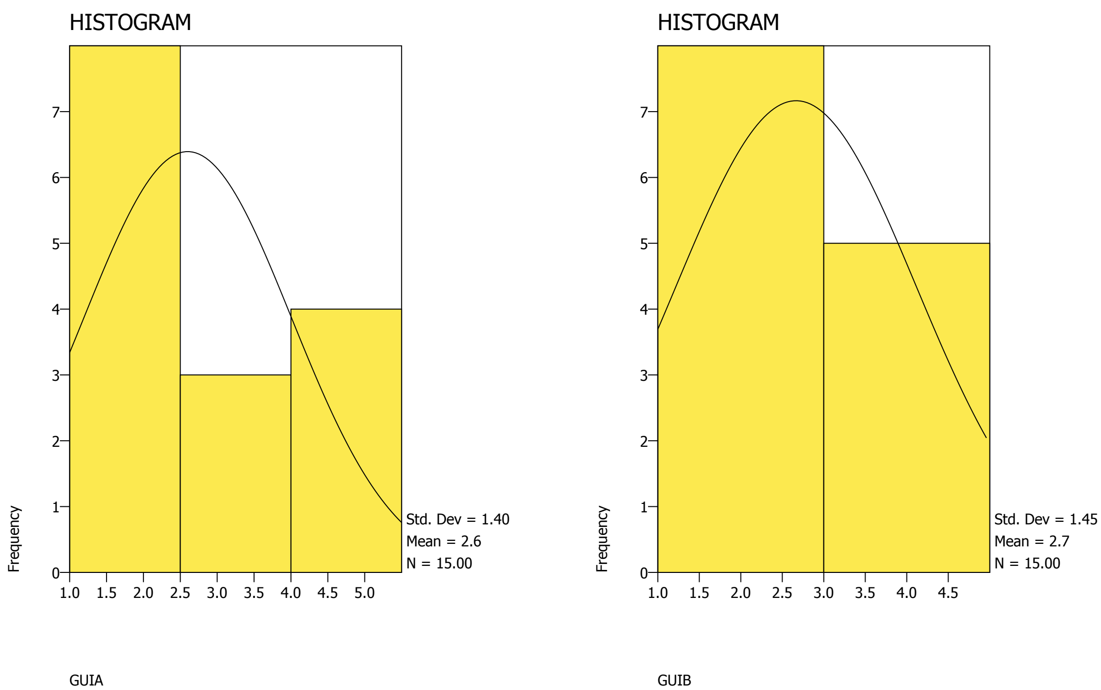


Figure 2.13: Likert scale analysis for both graphical user interfaces with average values and standard error bars.

	GUI A	*GUI B*
Mean	2.6	2.666666
Variance	1.97142857	2.09523
Observations	15	
Pearson Correlation	-0.31630588	
Hypothesized Mean Difference	0	
df	14	
t Stat	-0.11160428	
P(T<=t) one-tail	0.45636074	
t Critical one-tail	1.76131014	
P(T<=t) two-tail	0.91272148	
t Critical two-tail	2.14478669	

Figure 2.14: T-test for Likert scale analysis using Microsoft Excel.



(a) (b)

Figure 2.15: Histograms of data sets for Likert scale analysis with superimposed normal distribution curve.

		GUIA	GUIB
N		15	15
Normal Parameters	Mean	2.60	2.67
	Std. Deviation	1.40	1.45
Most Extreme Differences	Absolute	.20	.21
	Positive	.20	.21
	Negative	-.13	-.15
Kolmogorov-Smirnov Z		.77	.82
Asymp. Sig. (2-tailed)		.594	.518

Figure 2.16: One sample K-S test for Likert scale analysis for both graphical user interfaces.

Paired Sample Statistics

		Mean	N	Std. Deviation	S.E. Mean
Pair 1	GUIA	2.60	15	1.40	.36
	GUIB	2.67	15	1.45	.37

Paired Samples Correlations

		N	Correlation	Sig.
Pair 1	GUIA & GUIB	15	-.32	.251

Paired Samples Test

		Paired Differences							
					95% Confidence Interval of the Difference				
		Mean	Std. Deviation	Std. Error Mean	Lower	Upper	t	df	Sig. (2-tailed)
Pair 1	GUIA - GUIB	-.07	2.31	.60	-1.35	1.21	-.11	14	.913

Figure 2.17: T-test for Likert scale data set using PSPP.

Conclusions

Presented work described the process of designing, with preliminary steps, and evaluating a user interface for a specified task of motion analysis by a therapist.

The design has underdone several changes in the first and second iteration during and after the *"Sketch"* and gaining experience with the designer software *"EyesWeb"* with encountered *"Limitations"*. Therefore it is important that the initial design of the user interface is performed using low cost resource, like a simple sketch with a pencil on paper. This is because using a link in the chain of software and tools used for the design can hinder the creativity of the designer. Also, a detailed graphic design at an initial stage can distract the designer from the actual objectives of the application.

The designed user interface was based on a user centered design, where a deeper task, environment and user analysis took place in order to specify the requirements on the UI more precisely and omit design errors. It has been shown in *"Application and user analysis"* that even with little information about the target user, a quite sophisticated approach is possible. Where data is missing is can under circumstances be extrapolated as seen in *"User analysis"*. However even better results can be expected when already during this initial stage of the design process the end users participate. A therapist and a professional dealing with video processing would have been able to provide much more insight in the application design. This is not limited only to the features, but also includes terminology used in the design that should be greatly appreciated by other professionals from the same field. Unfortunately, medical expertise was unavailable during this development.

From the evaluation of the designed user it can be stated that the statistical tools help analysis once the interface has been developed, while laws involved in design of user interface are helpful before and during the process of user interface design. Concepts such as Fitt's law, Hick-Hyman law involved in design of user interfaces are important and have been taken into consideration.

Using carefully selected widgets in the UI is beneficiary for both the designer and the user. For the GUI designer, widgets offer design patterns that are readily available and appropriate for most of the tasks. For the end user, widgets are familiar and popularly used. Users find the related tasks then easy to understand and execute. Thus they make the interface more user friendly and less foreign.

Even though the layout of the two GUIs is very different, they both offer similar ease of navigation and time to perform a set of actions. Evaluation through statistical tools played a significant role in this aspect. Just visual analysis can mislead in conclusions and result in a incorrect decision to use a certain graphical user interface or not.

Mere objective measurement was not enough for the performed evaluation, therefore a subjective measurement, which depends on the emotional feedback of the user, was helpful in order to receive a more complete analysis. In the presented case, the two interfaces performed similarly according to the analysed measurements, however including more participants in the experiments should further confirm the results. There might not be a significant change in the mean time to perform the task, but the subjective judgement of the ease of navigation might significantly change depending on the test group. This can help determine, which interface is more user friendly based on the aspect of ease of navigation and allow for further improvements.

Next step would be to perform subsequent iteration in the user centered design to further improve the interface.