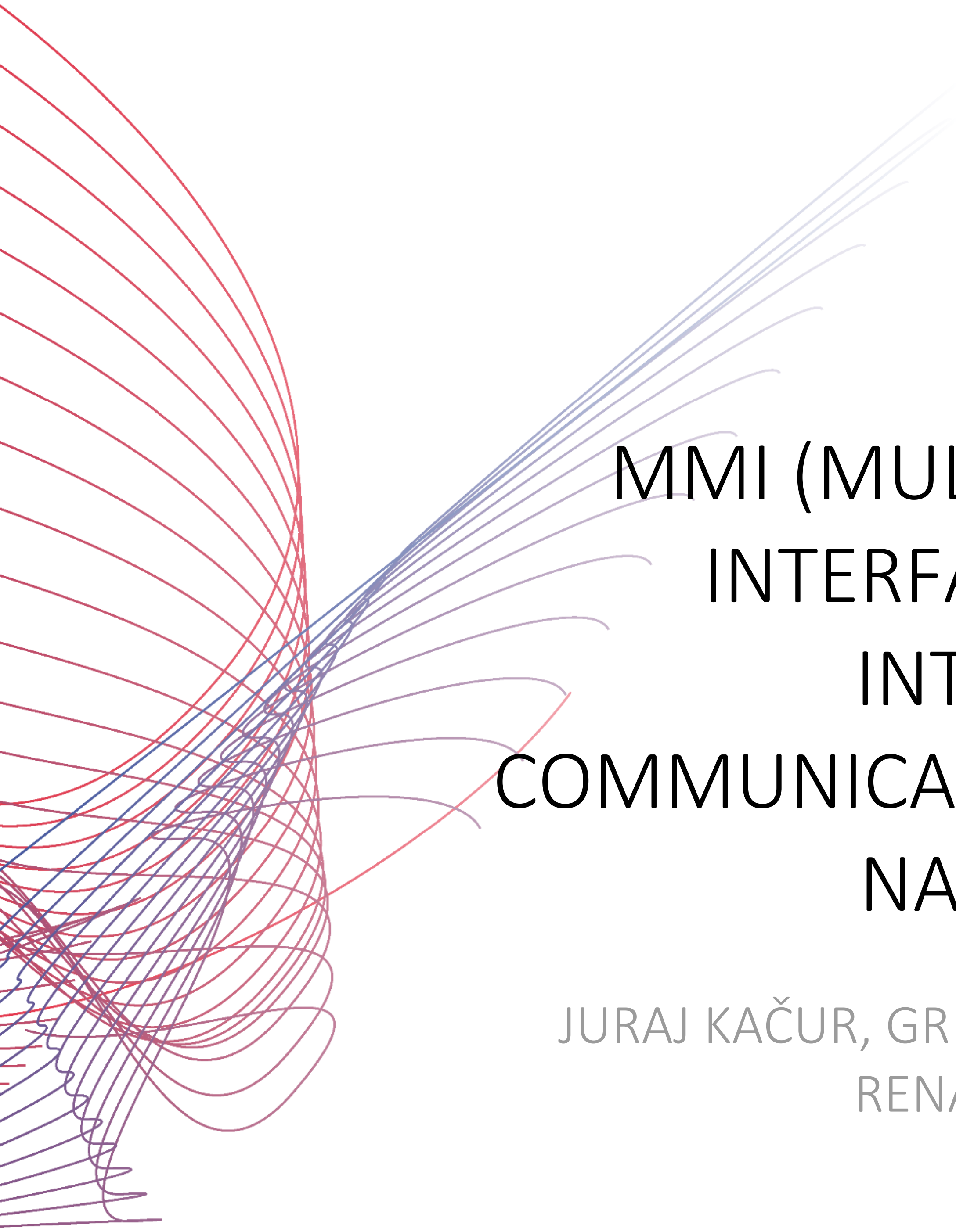
TECHpedia



MMI (MULTIMODAL INTERFACES) FOR INTERACTIVE COMMUNICATION AND NAVIGATION

JURAJ KAČUR, GREGOR ROZINAJ,
RENATA RYBÁROVÁ

Title:	MMI (multimodal interfaces) for interactive communication and navigation
Author:	Juraj Kačur, Gregor Rozinaj, Renata Rybárová
Published by:	Czech Technical University of Prague Faculty of electrical engineering
Contact address:	Technicka 2, Prague 6, Czech Republic
Phone Number:	+420 224352084
Print:	(only electronic form)
Number of pages:	39
Edition:	1st Edition, 2017
ISBN	978-80-01-06242-5

TechPedia
European Virtual Learning Platform for
Electrical and Information Engineering

http://www.techpedia.eu

Erasmus+

This project has been funded with support from the European Commission.
This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Explanatory Notes

$E=m\cdot c^2$	Definition
⚑	Interesting
i	Note
🧪	Example
🧩	Summary
+	Advantage
−	Disadvantage

ANNOTATION

Multimodal interface (MMI) consists of several topics and modules which serve for natural and user-friendly communication with the system. Altogether, these modules represent the functionality of MMI. The list of modules which can be part of the multi-modal interface is following: speaker recognition, speech and command recognition, multi face recognition, gesture navigation, eye navigation and speech synthesis. This module will introduce some of them, to help understand basic principles and algorithm which meet us in our daily lives.

OBJECTIVES

The main goal of the module is to introduce a student to the fundamental of multimodal interface and application. The student is clearly acquainted with the base principles of face recognition, speaker identification, voice commands navigation and gesture navigation. In the end all of mentioned modalities are described in possible multimodal applications.

LITERATURE

[1] Probst, Michael; Schumann, Sebastian; Rozinaj, Gregor; Minarik, Ivan; Rybárová, Renata; Oravec, Miloš. EVALUATION: Final Multimodal Interface for User/Group-Aware Personalisation, Deliverable 5.5.1, available at http://www.hbb-next.eu/index.php/documents, Decmber 2013.

[2] Vančo, Marek; Minárik, Ivan; Rybárová, Renata. Evolution of static gesture recognition. In: Redžúr 2014 proceedings; 8th International Workshop on Multimedia and Signal Processing; 13 May 2014, Dubrovnik, Croatia. Bratislava: Nakladateľstvo STU, 2014, p. 41-44. ISBN 978-80-227-4162-0.

[3] Rautaray, Siddhart S., Agrawal, Anupam. Vision based hand gesture recognition for human computer interaction: a survey, Springer, 2012, pp 106-113. ISBN 978-3-642-27386-5

[4] Kristensson, Per Ola; Denby, L.C. Continuous recognition and visualization of pen strokes and touch-screen gestures. In Proceedings of the Eighth Eurographics Symposium on Sketch-Based Interfaces and Modeling, 2011, pp. 95-102.

[5] Gülbahar, Mark; Probst, Michael; Schumann, Sebastian; Rozinaj, Gregor; Minarik, Ivan; Rybárová, Renata; Oravec, Miloš. Design and Protocol: Intermediate Multimodal Interface and Context Aware Recommendation Engine, Deliverable 5.3.1, available at http://www.hbb-next.eu/index.php/documents, October 2012.

[6] T. Kinnunen, H. Li, An overview of text-independent speaker recognition: from features to supervectors, Speech communication, Vol. 52, pp. 12-40, Elsevier, 2010

[7] Benesty J., Sondhi M. M., Huang Y., Springer Handbook of Speech Processing, Springer-Verlag, ISBN 978-3-540-49125-5, 2008

Index

1 MMI architecture

Currently, the most widely used interfaces for human–computer communication are keyboard, mouse, or touch tablet. These devices represent human's adaptation to computer limitations rather than the natural communication with computer. In the last few years a requirement began to pop up that humans need to communicate with machines in the same way as they do with each other: by speech, mimics or gestures, since they conceive much more information than peripheral devices approach. This leads us to the term *multimodal interface* (**MMI**).

Multimodal interface consists of several topics and modules which serve for natural and user-friendly communication with the system. Altogether, these modules represent the functionality of MMI. These modules can be part of the multi-modal interface:

- Multi voice identification
- Speech and voice command recognition
- Multi face recognition
- Gesture recognition and navigation
- Eye navigation
- Speech synthesis
- Recommendation engine

The general architecture of multimodal interface consists of several layers. Physical layer represents hardware input and output devices which enables interaction with real-world. Multimodal data provided by input devices (camera, sensor, microphone, etc.) are processed in parallel by each module separately. The MMI controller collects output data from all modules, evaluates and combines it into one output data stream. The stream contains information about recognized users and their requested actions.

2 User identification

Nowadays information technologies are getting more and more into the foreground. We meet with them in our daily lives, starting with smart-phones and ending with more sophisticated devices.

Every device working with our personal data should be therefore protected using some type of security system. The most widely used authentication type is password, but with development of information technologies and security protection algorithms, systems and applications start to use authentication based on the biometric factors. There are several benefits based on using biometric signals: biometrics are unique to a person, so it’s sufficient a person’s presence in front of the sensor and there is no need to remember several passwords or PINs. We know identification systems based on several different biometric modalities like iris, fingerprint, voice, face etc. In this part we will focus on the face recognition systems and speaker identification systems in general.

2.1 2D face and 3D face recognition

Face recognition is a subset of large field of pattern recognition research and technology. Human face recognition has become one of the most important biometrics authentication methods in the past few decades, due to its potential for a wide variety of applications and areas (surveillance, home security, border control etc.). Biometric systems for personal identification, which are developed by several vendors, achieve very high face recognition accuracy. The most of these applications require [1]:

- recognition systems which can recognize several faces from one video frame or one image
- high recognition rate
- illumination invariance
- stability under changing of face expressions and pose
- recognition in real time, etc.

+ The main advantage in comparison with other approaches is that the face recognition does not require any voluntary action by the user since face images can be acquired from a distance by a camera. Next advantage is that the acquisition devices are cheap and are becoming a commodity.

– The main drawback of the face recognition is its current relative ease with which can be defeated.

However, in comparison with speaker recognition, face recognition achieves much better results. In general there are three main approaches based on type of data which are used in the recognition process. We know methods based on 2D intensity image, 3D facial data and the technique which is using both types of data. The whole process of recognition consists of 3 main stages. The first one is acquisition and pre-processing, the second one is data registration and the third stage is recognition. Detail description of 2D and 3D face recognition can be found in module *UserIdentification*.

In modern systems multi-face recognition can be implemented. In that case all faces in the images are detected. Positions of detected faces are associated with the image. Next, the image is split in to several samples based on the positions of detected faces. These generated samples represent all faces in the scene and allow to track each face separately. The image split is a main step towards multi-face recognition within the systems.

2.2 Speaker identification

Generally speaking the task of speaker identification it to automatically decide who the unknown speech sample belongs to. The decision is based on a set of users stored in a database during a training phase. However if the observed decision confidence is too low the system may not recognized anybody.

As there is a fixed set of user who may be recognized at a time, this task is often called as a closed group problem. Speaker identification has been under serious scientific investigation for over 40 years and there is still and even grooving scientific effort going on. With the arrival of new and wide spread technologies it is finding grooving applications in many areas, just to mention few of them:

- **Forensic science**
- **Automatic indexing of speech and audio recordings stored in databases**
- **Natural and non-invasive method for accessing and protection of data and service,**
- **Applications for game industry**
- **Aids for disabled people**

Speaker identification is quite a tricky problem because of many reasons. Just mention some of them:

- **Speaker acoustical variability**
- **Modifications caused by current health, physical and mental state**
- **Presence of background noises**
- **Influence of a recording device and place of recording (echoes)**
- **Easy to replicate by recording devices**

However using speech as biometric signals has following advantages:

- **No need to use passwords**
- **Non invasive method**
- **An easy data acquisition method for obtaining a biometric signal**

There are many applications for speaker identification that exhibit different levels of complexity, requirements, confidentiality, response time, etc. Thus we distinguish several major classes of identification systems:

- **Text independent systems** - works regardless of what was said, thus there are no requirements on the phrases, sentences or even words to be said, perhaps except the length of a genuine speech, i.e. not corrupted speech without pauses. The accuracy range of such systems is somewhere between 80 to 95%. Currently they are the most frequent ones as they imply no or very few restrictions. However, because their lower accuracy they can't be used in security critical applications.

- **Text dependent systems** - require precise phrases to be uttered by an unknown individual. As such systems may utilize special features of individuals while pronouncing certain phrases (so called co-articulation effect) the systems naturally exhibit higher accuracy rates, ranging from 95% to even 99%. Furthermore these systems can be even more accurate and robust if the uttered text is a password specific to every user. In specific cases those systems may alter the required text to be said by asking different questions.

- **Acoustic information based systems**- utilize only short time acoustic information derived from the speech. It is mostly sort of modified and filtered spectrum that captures number, position and widths of major formant frequencies. They form the basis for all systems as they contain information about physical properties of vocal organs (sizes, shapes, weight, toughness, etc.) of each individual.

- **Prosody information systems**- evaluate except acoustic information also prosodic information that reflex individual speaking habits like rhythm, dynamic, speaking pace, mood, voice modulation etc. These features are observed and derived from longer time periods.

- **Higher level feature systems**- incorporate also features like dialect, used vocabulary from which a social status can be inferred, etc. These features however do not provide enough discrimination thus they can't be successfully used without other features (acoustic and prosodic).

- **Real time systems-** provide instant information about the speaker's identity; however certain delay of several seconds is needed for reliability.

- **Off line systems-** can use more complex algorithms to provide as precise decision as possible while using a whole speech recording.

The above mentioned statements and ideas can be further unfolded into many areas and concept spreading over several branches of science. Thus for more thorough introduction to speaker identification problem please refer to e.g. [6].

2.3 Multi-level identification

Some applications (e.g. AppStore) can be accessed by multiple users from one or from different type of devices. For this purpose a multi-level identification is introduced.

The multi-level identification deals with varying confidentiality requirements (like different access rights or control) through multiple biometric modalities in combination with password and PIN methods. From group of biometric modalities voice and face recognition is used to identify the user. Depending on the outcome of this process, it might allow access to the system or application or request the person to enter another credentials — a username, password or PIN. In the future, systems can be extended to use iris recognition and replace the PIN or password method and thus improve security.

3 Communication via speech commands

Communication with machines via speech commands falls into a vast research domain called *automatic speech recognition* (**ASR**).

This term generally means automatic (by a machine) transcription of a spoken language. The input is a speech signal mostly present in a digital form as a sequence of numbers. The output is a text in the form of strings of words that exist in a vocabulary (as it can be rather extensive usually only restricted vocabulary is used for a particular domain). Furthermore the output string follows either regular grammar rules or frequent patterns that can be observed in spoken language (statistical language modeling). Thus the task is often called as speech to text problem.

ASR should be distinguished from the task of understanding of what was said which operates on higher level (the input is a text) and the branch of science that deals with this problem is called an artificial intelligence.

For several decades there has been a big effort going on to construct an ASR system that could be widely used especially in the following areas: information retrieval systems, dialog systems, aids for handicapped people, etc. But it was only recently since domain oriented applications came out of laboratories. Currently as the technology and knowledge made crucial steps other more sophisticated applications like dictation systems or even automatic transcription of natural speech are emerging. It is because the task is so complex facing many obstacles that must be solved by different domains of science. Practical systems for general public should operate in real and mostly very adverse conditions (great variability of noises, recording devices, employment places, etc.), must accept great variability of spoken language (rules are rather lose), speaker variability (in term of an acoustic form), huge vocabularies, just to mention few of them. Furthermore a general user requires an immediate system response (working in the real time), he is not willing to change his speaking habit or to restrict his vocabulary and quickly looses his patience if the system doesn’t work with a high accuracy.

As the range of possible ASR applications is quite wide and so is the complexity of the systems required for solving particular tasks (actually it is growing even much faster with the growing requirements) there exist several classes of systems according to which the ASR systems are classified. The main criterion related to ASR classification is based on the vocabulary size, and we distinguish:

- **Small-** several dozens or even hundreds of words
- **Medium-** several hundreds or even thousands of words
- **Large-** several tens or hundreds of thousands of words

However these numbers change as the technology progress.

Then systems can be speaker dependent or speaker independent that means whether they can equally operate regardless of who is speaking. Further, it is important if the system provide an immediate response or works off line, thus we have real time systems or off line system. Next it is quite vital to know in which form they expect the speech samples to be processed. Thus we distinguish:

- Isolated word recognition – where a single word from a dictionary is expected
- Dictation systems **-** require string of words that are well separated so they can be processed on individual bases
- Continuous speech **-** that however follows more or less severe limitations given by some sort of grammar
- Natural speech**-** is the most challenging situation that encompasses both continual speech with no or very loose limitations.

Finally we can classify systems based on what speech units they model (phonemes, word, syllables, phrases …) and what models they use, e.g. statistical modeling like *hidden Markov models* (**HMM)**.

A speech signal is produced by human vocal organs and is observed through air vibrations. Except many other it carries lexical information (what was actually said) that is represented as a sequence of different acoustic sounds. Set of basic sounds that are called phonemes is used to build words of a particular language. Their number may vary among languages (usually from 40 to 60). However, their acoustic forms differ from speaker to speaker and neighboring phonemes influence each other (co-articulation phenomenon).

To suppress all information that is irrelevant or may even hinder correct speech recognition should be removed prior to the recognition process. Basically we are interested only in the lexical information so the remaining one like speaker identity, mood, actual health condition, dialect, speech impairments and habits should be suppressed. It is partially the task of a speech extraction method that is to pick up only the needed information that is to be processed by following blocks. There is approximately 10b/s lexical information stream while the bit rate for speech signal is about 100kb/s. Thus the extraction method can be looked at as a bit rate compressor.

The aim is to simulate the auditory system of humans, mathematically describe it, simplify for practical handling and optionally adapt it for a correct and simple use with the selected types of recognition and classification methods.

There are several proper feature extraction methods that either simulates the speech production process or they mimic a human auditory system (critical bands). It is so because the auditory system has evolved just to focus on relevant information suppressing ubiquitous noises and distortions.

Through years of the research it was discovered that except time domain the most significant discriminative information to classify phonemes between each other lays in the frequency domain. More precisely, it is located in the position and shape of dominant frequency components. To demonstrate this, in Fig. 3.1 there is a frequency spectrum for a vowel “e”, its magnitude envelop with depicted formant frequencies (major spectral peaks). In Fig. 3.2 the time representation of the corresponding vowel ‘e’ is given.

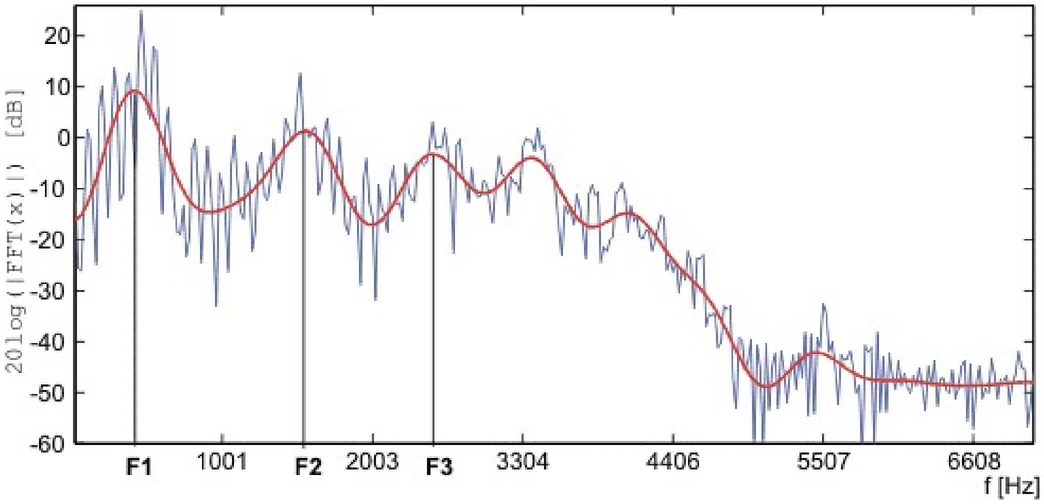


Figure 3.1. A spectrum, formant frequencies and a spectral envelope for a vowel “e”.

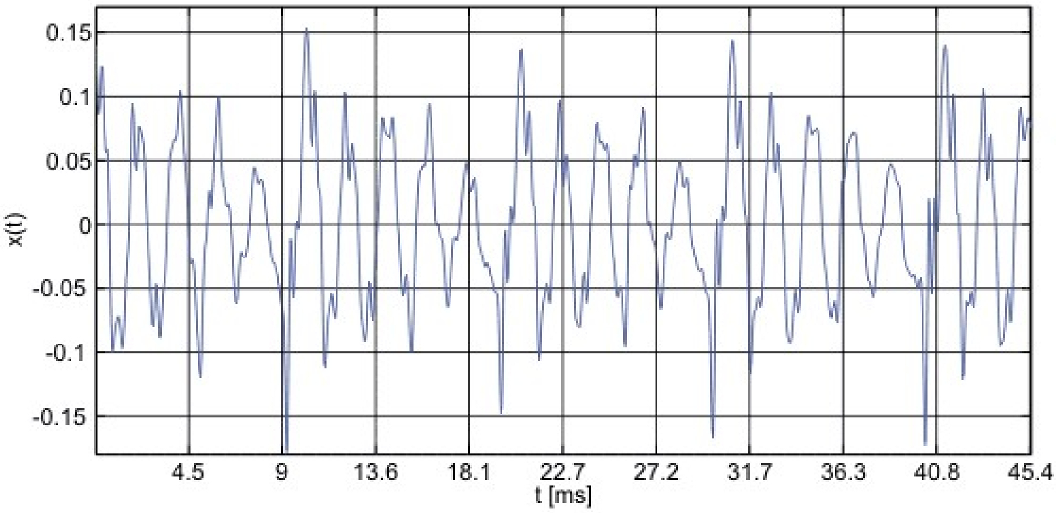


Figure 3.2. A signal of a vowel “e”.

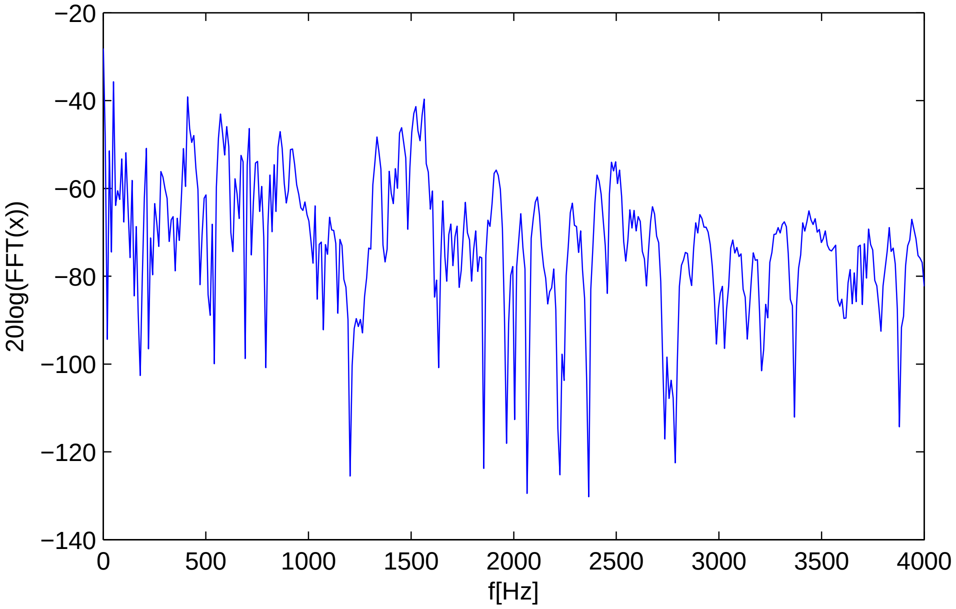


Figure 3.3. A spectrum of a phoneme “t”.

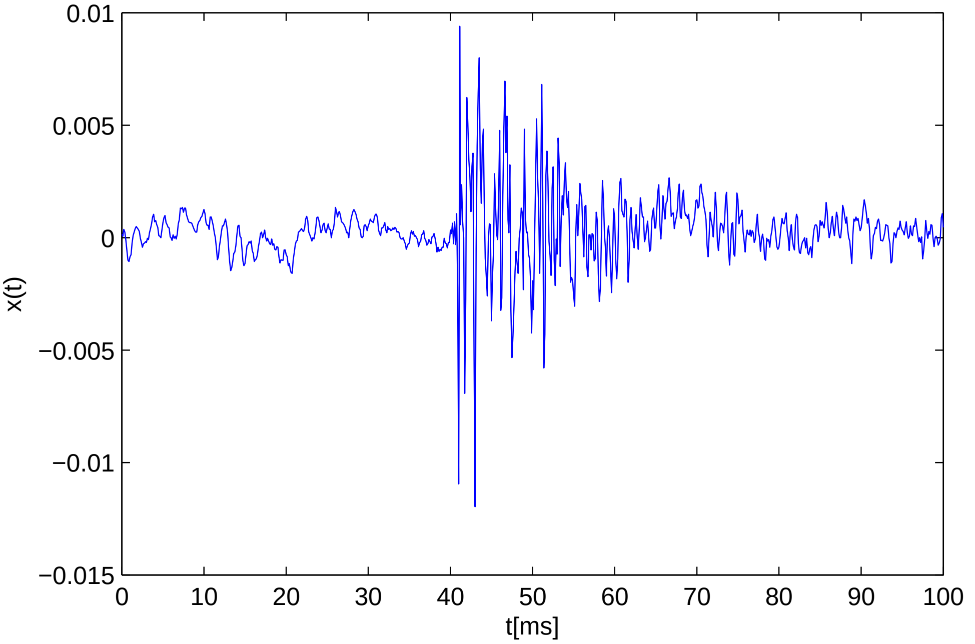


Figure 3.4. A signal of a phoneme “t”.

To see the time and frequency differences between various phonemes, in Fig. 3.3 there is depicted the spectrum and in Fig 3.4 the time course of a phoneme “t”. To sum it up, in the following table first two formant frequencies for vowels are listed separately for males and females (average figures gathered over population). These positions provide a very rough but a simple way how to classify phonemes based on their spectral representations.

Table 3.1. First two formant frequencies observed in common vowels for males and females

vowel	Males		Females	
	F1 [Hz]	F2 [Hz]	F1 [Hz]	F2 [Hz]
a	730	1100	850	1200
e	530	1850	600	2350
i	400	2000	430	2500
o	570	850	590	900
u	440	1000	470	1150

Thus extracted features should be able to estimate and discriminate those differences in formant frequencies. On the other hand they should neglect variations that are natural and inaudible. In the following table the most relevant audible and inaudible spectral modifications are listed.

Table 3.2. Audible and inaudible modifications of speech spectra

Sort of modifications	
Audible	**Inaudible**
Number of formant frequencies	Overall tilt of the spectra
Position of formant frequencies	Frequencies under the first formant frequency
Widths of formant frequencies	Frequencies above the 3rd format frequency
-	A narrow band stop filtering

Moreover intensity of signals is perceived non-linearly that can be approximated by a logarithm function.

From the noise point of view features should be insensitive to additive and convolutional noises. Last but not least a good feature must be easy to implement, mathematically tractable, and has a compact representation. Usually it is beneficial if features are linearly independent from each other that ease their subsequent processing.

3.1 Acoustic features for speech recognition

First it should be noted there is no feature that would completely fulfill all the statements mentioned above. Therefore the research is still active and many acoustic speech features have been designed but the most commonly used are *Mel frequency cepstral coefficients* (**MFCC**) and *Perceptual Linear Prediction* (**PLP**). PLP and MFCC try to simulate the human auditory system which results in good performance in speech recognition tasks. Both are able to capture positions and widths of formants that are most perceivable. Despite obvious similarities they differ in the psychoacoustic phenomena they encompass.

MFCC applies high pass filter (suppression of the lip radiation), segmentation of speech by Hamming window that is followed by a conversion to the spectrum by DFT. Next the spectrum is non-linearly warped into the Mel scale (psychoacoustic scale that reflects human perception) over which equally spaced triangular windows with 50% overlap are placed to simulate a filter bank (see Fig. 3.5). In the final stage of the calculation logarithm and **Discrete Cosine Transform** (*DCT*) transforms are applied. Moreover DCT suppress dependency among features.

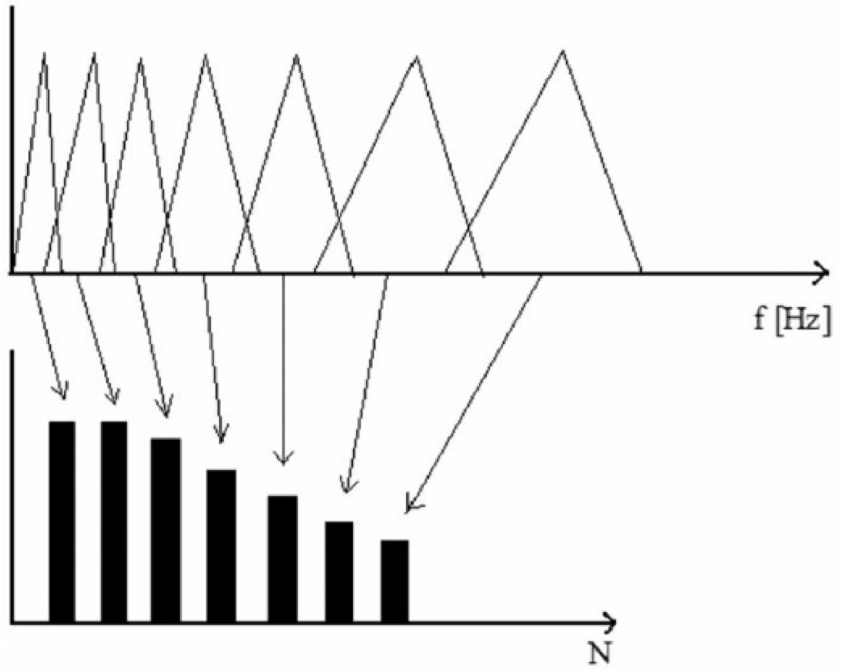


Figure 3.5. A spectral division to bank of filters using Mel scale and triangular windows

PLP features differs from MFCC in several aspects as follows: the usage of Bark frequency scale, smoothing and sampling the bark-scaled spectra in 1 bark intervals, equal loudness weighting, transformation of energies into loudness, calculation of a linear speech production model, and its transformation into a cepstrum.

Thus PLP applies more complex psychoacoustic processing than MFCC; however both of them usually produce similar results for speech recognition and laboratory conditions.

Dynamic features and the energy

Speech is basically a particular sequence of different sounds thus it makes sense to measure and evaluate proper transitions between them. Most common method to do so is via delta and acceleration coefficients constructed over acoustic features in the time. They can be computed as differences between two adjacent frames or in more general case as a linear combination of differences covering a wider time span. Furthermore, it was shown that an energy envelop can locate the position of high-energy vowels and low-energy unvoiced consonants that augments the overall discrimination information. So the energy (normalized) feature is often added to the acoustic features as well.

3.2 Speech recognition methods

Once proper speech features have been extracted the speech is in the form of a sequence of feature vectors, e.g. MFCC. Then the process of recognition roughly said takes samples or models of known speech units (phonemes, words, etc. from a training database) and compares them with the unknown speech sample, i.e. its feature vectors. Then the sample or model with the highest match (score) is claim to be the recognized word. Because of the special properties of speech signals, namely: every pair of unique signals of the same word differs by length (somebody speaks faster or slower, etc.). Moreover this length variability is not uniformly distributed along the time axis, thus some parts may last longer while the other may be uttered faster. Therefore the basic approach to solve length differences by a linear interpolation or decimation can't be successfully applied here. Furthermore, based on the models the system uses it is usually necessary to concatenate a sequence of samples or models to represent certain word or even a whole sentence. These two main phenomena (non-uniform variability in length and concatenation of models) specific to speech gave a rise to the development of specific classification approaches. Currently the most common are *Dynamic Time Warping* (**DTW**) and *Hidden Markov Model* (**HMM**) methods. However there exist more modifications or even combinations that may be eligible in particular applications. Thus in the following a brief introduction of those two methods is provided.

DTW

DTW is an abbreviation to dynamic time warping which is a method that acoustically compares sequences of two speech feature utterances (reference and the test one). It is based on nonlinear time warping during the comparison process so that these two sequences are as close as possible (evaluated by a proper acoustical measure). Thus it compensates for nonlinear variations in lengths within words.

To do so the first and last vectors in of the two sequences must be aligned. Therefore this approach requires the knowledge of word boundaries in prior, which may be a tricky task by itself while it is to be done automatically. However there exist modifications to DTW that relax this strict requirement.

The method briefly tries to find a mapping between two sequences of vectors of different lengths so that each vector has a partner vector from the other sequence to be compared with. This means some vectors at particular time may be omitted or one vector can be mapped to more vector from the other sequence. Of course this process can be done arbitrary so this mapping must follow certain logical limitations i.e. beginning and end vectors of a sequence must be mapped to their counterparts in the second sequence, the warping functions must be non decreasing, there is a maximal allowed discrepancy that this nonlinear mapping can overcome (usually vectors whose indexes are more than twice of each other can't be compared, etc.). In the DTW calculation process two matrices are used (local distance and global distance). In the local matrix there are stored acoustic distances between reference and unknown feature vectors. The global one is used to calculate the path (mapping) and to accumulate the minimal distance along the optimal path.

Thus there is a minimal distance and an optimal path related to any element of the global matrix that connects its position to the beginning point which is in the left right corner. This situation is depicted in Fig. 3.6. Of course, there are natural limitations on directions how to move from one point to subsequent ones (non-declining in both horizontal and vertical directions). Once the search process reaches the end point of the global matrix (upper right corner) the comparison is over and the distance is found. This process repeats for every word from a dictionary and the word with the least global distance is claimed to be the unknown one. As it can be seen this method is eligible for recognition of isolated words, or commands.

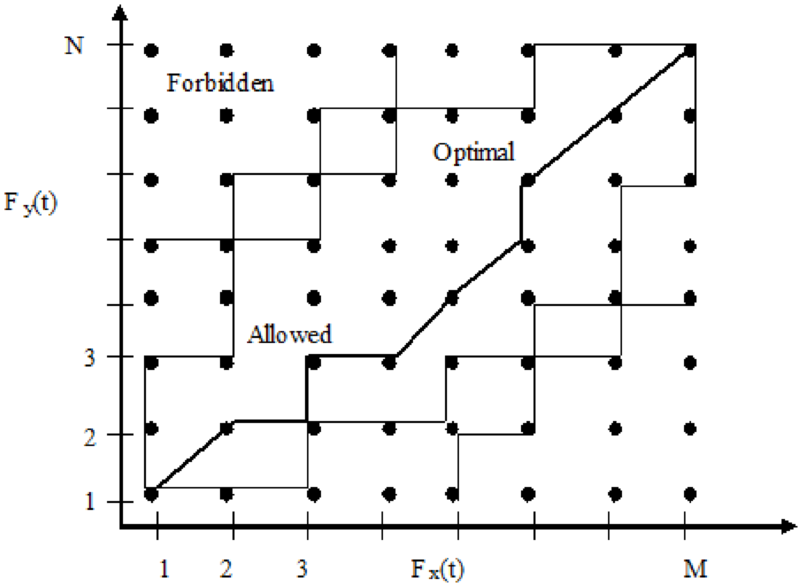


Figure 3.6. A global matrix with the optimal path and global path limitations.

DTW had a significant position in the speech recognition, especially for the problem of isolated word recognition that is speaker dependent. However, as the requirements were growing e.g. speaker independence and continual speech recognition, it was gradually losing its position to HMM method.

Hidden Markov Models (HMM)

Hidden Markov Model is a statistical modeling technique mainly used for speech recognition that solves both the speaker independence and concatenation of basic models (to form word, phrases, sentences and even covers continual speech) in a mathematically elegant way.

For each selected speech unit (phonemes, syllables, words, …) a HMM model is created having certain structure. Usually all models share the same structure and they differ only by free parameters of the model. In the training process just those free parameters are set using the training database. Training database consist of speech utterances that are labeled so that it is known what was exactly said.

Parameters of HMM models are mostly adjusted in such a way that the models describe the training data with the highest probability, the so called maximum likelihood criterion. However, some systems that use different strategies based on maximal separation between models or minimizing error rates (discriminative criteria) may provide more accurate results.

Each model consists of several states that are connected to each other. With each connection there is associated a transition probability (p). Further there is an initial probability (π) for each state which is the probability the model starts in the given state.

Having such model the probability of a state sequence *S₁, S₂, S₃,.. S_N* is given by:

$$P(S_1, S_2, ..., S_{N-1}, S_N) = \pi(s_1) p_{12} ... p_{(N-1)N}$$

Moreover there is an additional probability associated with each state and that is the probability of observing a feature vector *X* in a given state S i.e. *P(X/S).* Then the probability of observing sequences of feature vectors $X_1, ..., X_M$, and states $S_1, S_2, ..$ S_N is as follows:

$$P(S_1, X_1, S_2, X_2, ..., S_N, X_N) =$$
$$= \pi_1 P(X_1 / S_1) p_{12} P(X_2 / S_2) p_{23} ... P(X_{N-1} / S_{N-1}) p_{N-1,N} P(X_N / S_N)$$

An example of a 4 state left-right HMM model is shown in Fig. 3.7.

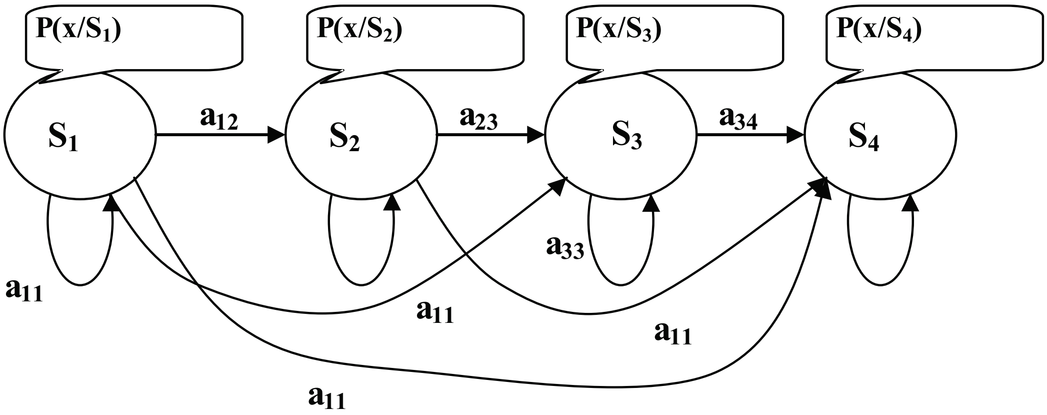


Figure 3.7. A 4 state left-right HMM model.

Then the recognition process calculates the probability of an unknown sequence on all HMM models in a dictionary and chooses the one with the highest probability. The process is schematically depicted in Fig.3.8.

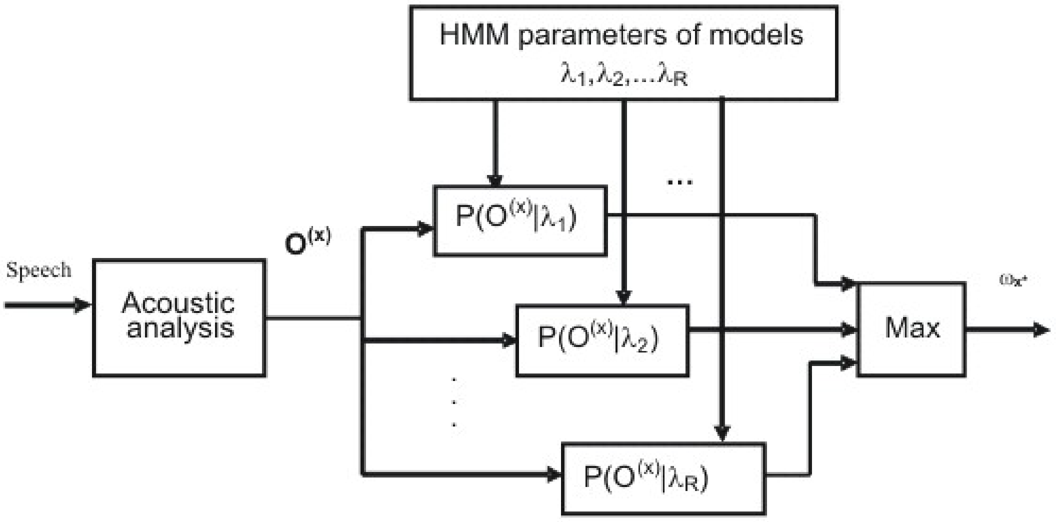


Figure 3.8. The speech recognition process based on HMM.

At presence the most advanced HMM systems achieving the lowest *word error rates* (**WER**) use different strategies like *Maximal Mutual Information* (**MMI**), *Minimal classification error* (**MCE**), etc. Both MMI and MCE belong to the discriminative training. Finally there are very successful hybrid connections with other classification methods like Support Vector Machines or Neural Networks providing even lower WER on unseen data.

More detailed description of the speech recognition technology and used methods can be found e.g. in [7].

4 Gesture navigation

Nowadays gestures are very popular way for application control and many people are using it every day. Actually, we can use gestures in our mobile devices, computers applications, game consoles, etc. It is expected that gestural interaction will be available in almost every device in just a few years' time. High popularity of gesture navigation forces researchers to improve these technologies. This is an obvious trend since computer performance is no longer the bottleneck of the more natural navigation and control using gestures [2].

There are two main areas in the field of gesture recognition depending on the input hardware. The first area focuses on gestures designed for touch devices, such as tablets, touchpads or smartphones. The other area deals with gestures performed in front of a 2-dimensional or stereoscopic camera. With some adjustments, the same algorithms can be applied to recognize gestures in both areas. Additionally, the algorithms for stereoscopic cameras can be modified to use 3D coordinates for tracking in all three dimensions [2].

4.1 Gesture classification based on various criteria

The gesture controlled application become more and more popular. The gestures can be divided into two basic categories by user experience. Innate gestures that are based on general experience of all users such as to move an object to the right by moving hand to the right, catch an object with closed fingers, etc. Naturally, innate gesture can be effected by habits or culture. By using these gesture we can support no need to learn user to get experience for gesture control. We suppose the user can control application naturally. Second group of gestures are learned gestures. In this case it is necessary to teach the users. The gestures used for navigation of the systems and applications has to be easy, natural and has to spend minimum of human energy.

Static gestures

Static gestures do not depend on movement. Static gestures represent shapes of the gesturing limbs, which carry the meaningful information (Fig. 4.1).

They are "non-moving" type of gestures where we do not need information about the motion. Thus there is no need to investigate a sequence of frames, rather than the actual image frame. A sequence of several frames containing different gestures is only important in the higher level of comprehension.

Continual gestures

Continuous gestures serve as a base for application interaction. Continual gesture is prolonged tracking of movement where no specific pose is recognized but the movement is used to interact with the application or virtual environment (Fig. 4.1).

The typical example of continual gestures is game control using a touchless technology such as Microsoft Kinect, etc. where the system maps the changes in postures to changes in the video game, but there is no specific movement to trigger a sequence of changes.

Dynamic gestures

Dynamic gestures represent movement that allows users to directly manipulate an object or control application (Fig. 4.1).

Dynamic gestures can be defined in two ways. First approach is to algorithmically set rules and conditions that the performed gesture has to pass to be evaluated and successfully recognized. The second way is to use templates. Templates are sets of points which identify the shape of the gesture. Sophisticated algorithms are used to evaluate success rate when comparing user movements with templates.

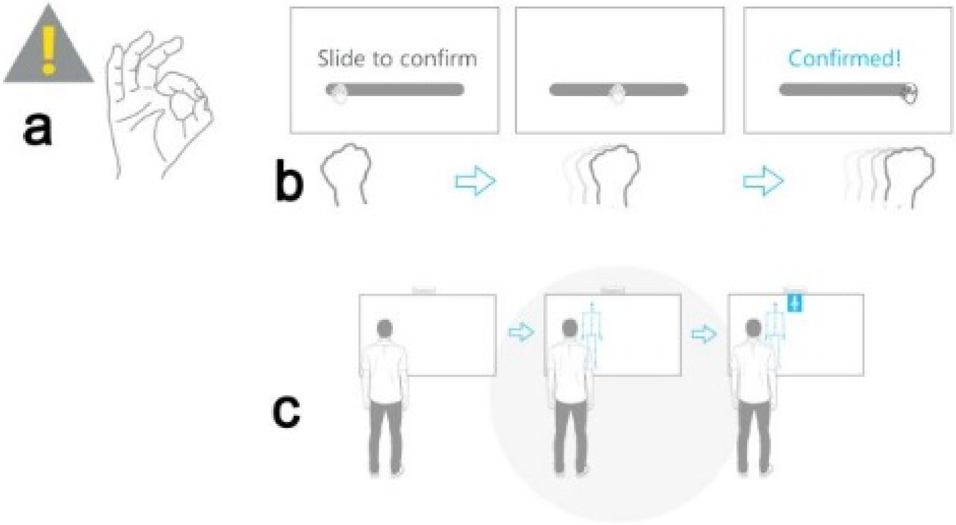


Fig. 4.1 – Gesture types: a) static gesture, b) dynamic gesture, c) continual gesture

4.2 Image data acquisition

In early research of gesture recognition algorithms design was based on **RGB** (*Red Green Blue*) camera input. This approach brings some advantages, which, however, are outweighed by disadvantages. RGB image consists of color channels and exist many factors which negatively influence quality of RGB image.

–

Illumination change can radically influence quality of image. A change in illumination conditions in a room absolutely changes the brightness and contrast of image. This makes recognition using RGB image uncertain [2].

In last few years the researchers started to use depth images for processing. Depth image is a two-dimensional image that contains additional information – depth. Most depth sensors are based on *infrared* (**IR**) emission (e.g. the Kinect sensor in Fig. 4.2). The sensor returns data which represent distance of each pixel in the frame from the sensor. The obtained distances can be simply transformed into grayscale representation and so depth data are represented as grayscale video sequence. Obtaining data does not require any special conditions.

+

The depth data have more advantages against RGB cameras. The main advantage is, that the depth camera makes use of infra-red light and in infra-red zone isn't so much noised. So infrared sensor isn't so sensitive than RGB sensor [2].

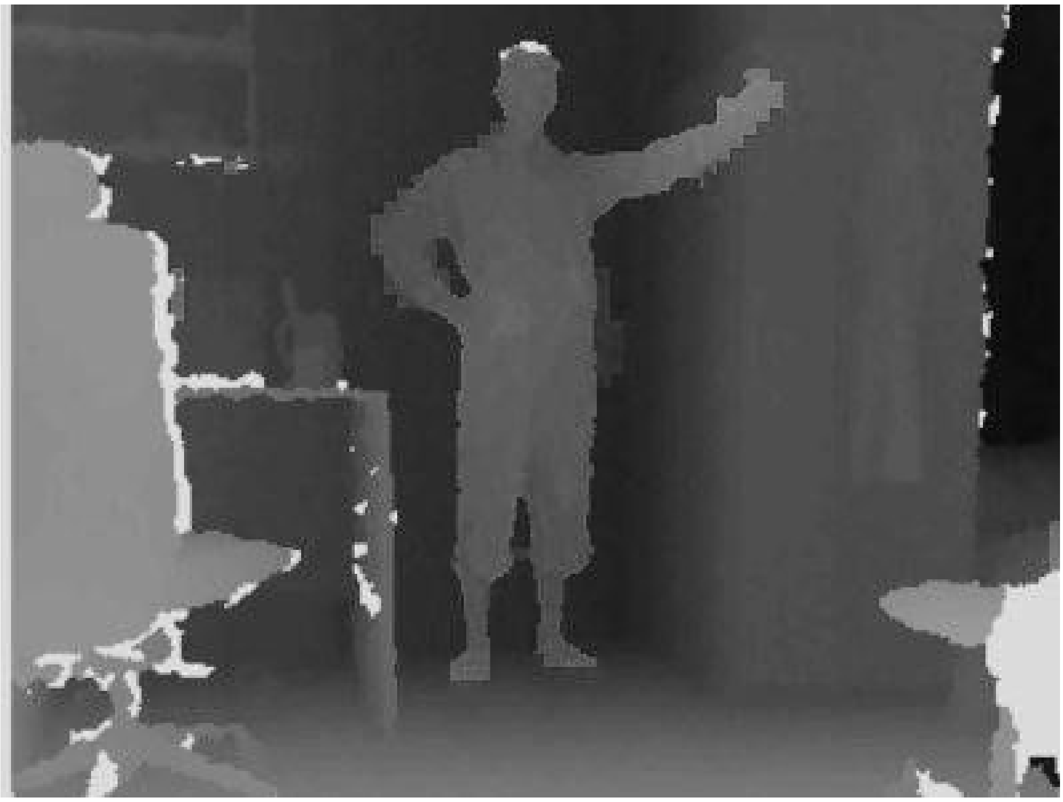


Fig. 4.2 – Image from depth camera

The distance of each pixel from the sensor is defined in millimeters.

If we want to convert depth image array into grayscale image, we need to know minimal and maximal possible distance.

$$\text{intensity} = 255 - \frac{255 - (d - d_{\min})}{d_{\max}}$$

where, *d* is actual distance of the given pixel, d_{min} is minimal possible distance from the sensor and d_{max} is maximal possible distance from the sensor [2].

To help with image data acquisition process some extra elements, like luminous bracelet, ring or special small ball in hand.

4.3 Gestures design methodology

The issue of gesture recognition is not trivial. All limitations and inaccuracies that occur when recording images using various devices have to be taken into account during process of designing gestures for application or system control. The methodology for gestures design has to fulfil several requirements: naturalness, easy to remember and comfort. Comfort is the most important criterion for gesture controlled application.

As in any new field also in gesture control one of the main problems in adaptation process is nonexistent standardization. When ten random people are asked to make a gesture to start the music, it is highly likely that at least 5 different gestures are proposed. It is therefore not possible to establish a solid set of gestures that suit to everyone. This is the reason why every producer providing gesture control uses a different set of gestures. It's difficult to practical use, since the user has to learn new gestures for any new system. The existence at least one standard set of gestures for controlling basic functions could significantly accelerate users to start using gesture control and putting into practice. Nowadays, some control gestures via television sets are offered, which are based on image sensing by 2D camera. This enables easy navigation through the menu of the TV. Despite its imperfections the educational effect on people who are gradually getting used to this kind of control is beneficial.

Also present gesture sets are based on peripheral input devices used with computers. They basically try to remove the device but keep the same usage patterns. In order to come closer to natural (touch-less) gesture-based operation the concept has to change so that gesture sets are designed from ground up, like if there were no peripherals other than sensors.

4.4 Basic hand gesture recognition techniques

The overall goal of hand gesture recognition is the interpretation of the following: the hand(s) location, posture, or gesture conveys.

Gesture recognition can be conducted in two manners. Either a data glove is used which transforms the body flexions into movement information, or vision-based approach is applied where a camera serves as a human eye to record body positions which are then extracted using image processing [5].

It is clear that the first method might bring precise results.

The vision-based approach, on the other hand, has no other equipment requirements for the end user (except for the camera), making it suitable for general applications [5].

Data glove is rather uncomfortable in terms of user convenience. Also, equipment needed to employ the method would be unacceptably costly for most of standard customers making it only suitable for special use.

The drawback of vision-based method, however, lays in algorithmic complexity where considerable amount of time and computing power is required to extract body movements [5].

There are various algorithms available which focus on different aspects of the gesturing person (and take different assumptions). Generally, they can be divided into two categories, appearance- and 3D model-based approaches. The 3D model-based approach compares the input parameters of a limb with 2D projection of a 3D limb model. The appearance-based approach uses image features to model the visual appearance of a limb and compares it with extracted image features from the video input [5].

In section 4.1 three gesture types were defined. A general classifier is used to detect static gestures (i.e. postures). Under classifier we can understand an element on which arrangement into group or category is being decided. So classifier tells us which gesture or pose was recognized. However, dynamic hand gestures have a temporal aspect and require techniques that handle this dimension, e.g. Hidden Markov Models (HMM). The other way is using motion based model.

Some of the techniques used for static (and dynamic) hand gesture recognition are K-means algorithm, KNN, SVM, already mentioned HMM algorithm, DTW algorithm or neural networks [3].

Techniques used for static gestures recognition

For faster data processing in gesture recognition process, the whole image area cannot be processed.

After the hand is automatically detected, the area around the hand is allocated which is then only processed to reduce the processing load. The obtained distances is converted into the grayscale image to get the contour of the hand.

Following two described methods for gesture recognition process.

Convexity defects

In the first step hand must be separated from the background. The separation can be provided by using depth information, determining which pixels of image belong to hand. Second base step is to detect contour of hand.

The shapes of many complex objects are well characterized by convexity defects. Fig. 4.3 illustrates the concept of a convexity defect using an image of a human hand. The convex hull is pictured as a dark line around the hand, and the regions labelled A through H are each "defects" relative to that hull. As it can be seen, these convexity defects characterize also the state of the hand. Algorithm returns the coordinates of three points, the *start point*, *deepest point* and *end point* (Fig. 4.4) of the defect, where the deepest point is understood as maximum distance between the hull and hand contour.

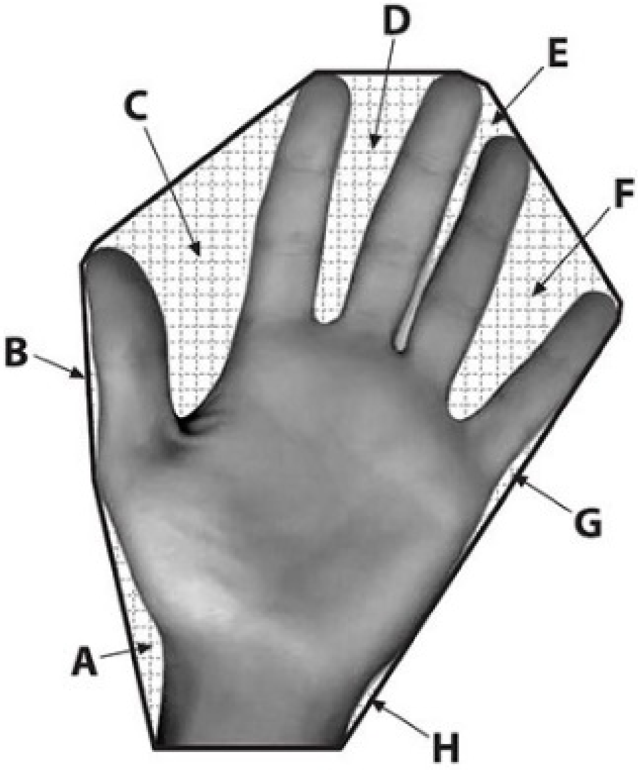


Fig. 4.3 – Convex hull pictured as a dark line around the hand

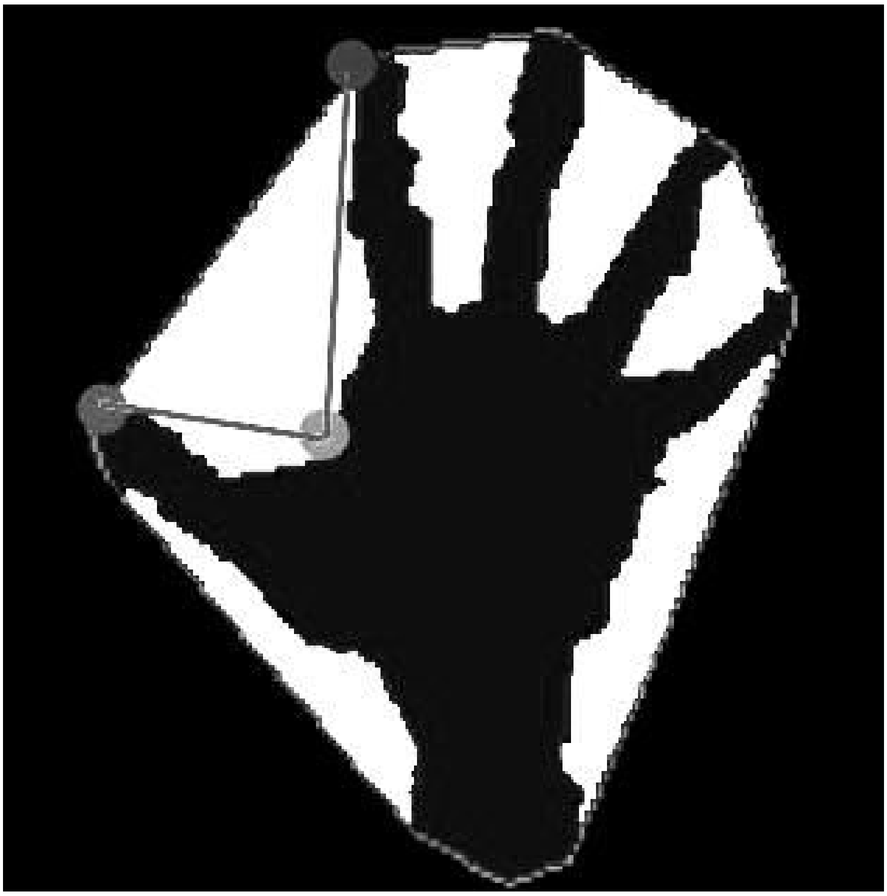
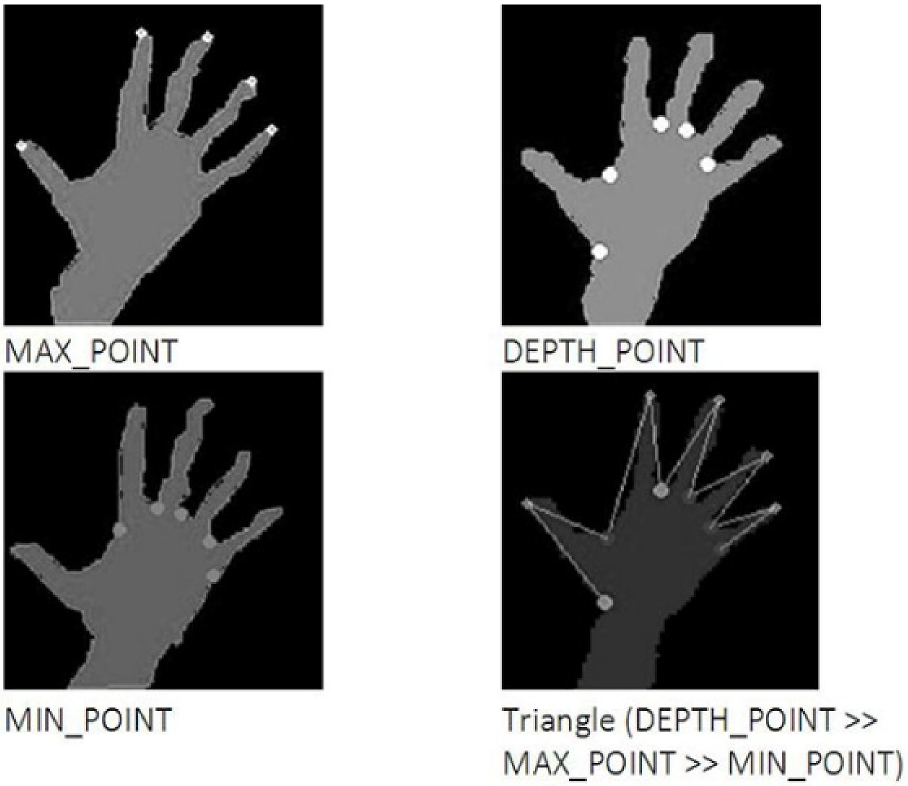


Fig. 4.4 – Marked areas are convexity defects and three points: start point (left), deepest point (bottom) and end point (top)

The goal of the algorithm is finding the point on the finger that is the farthest from the centre of hand. The first step is to remove all defects whose height is less than a specified value. Then we remove the defects that have a distance between starting point and maximum point more than a specified value, defined dynamically (if two points are too far, it means that they cannot represent a finger). Next also the defects that have a distance between start point and depth point less than a specified value are removed. This value changes dynamically according to the size of the region where the hand is detected (Fig. 4.5). The final step is to remove all the defects that occur below the wrist.



MAX_POINT

DEPTH_POINT

MIN_POINT

Triangle (DEPTH_POINT >> MAX_POINT >> MIN_POINT)

Fig. 4.5 – To illustrate the logic behind using triangles to eliminate unnecessary defects

Part-based hand gesture recognition

In the first step of this algorithm, data segments containing hands are obtained and converted into binary image. The centre of the palm (Fig. 4.6) is computed by inner circle and adding a point onto contour hull that belongs to hand and has maximal distance from the found defect.

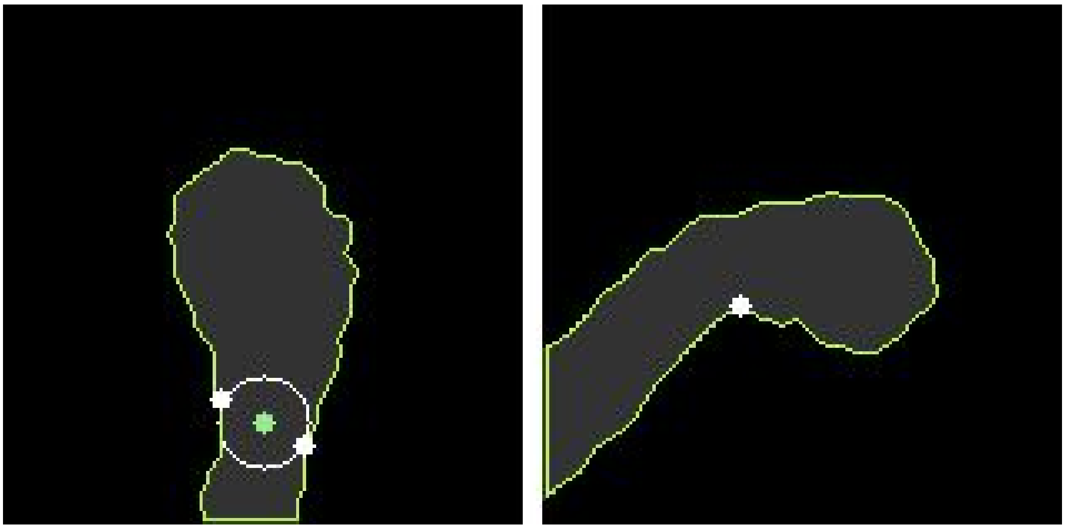


Fig. 4.6 – Finding the centre of the palm

All points of the hand contour are mapped onto X-axis. Y-axis then describes the relative distance of each point from the centre of the palm. All mapped points create a curve (Fig. 4.7).

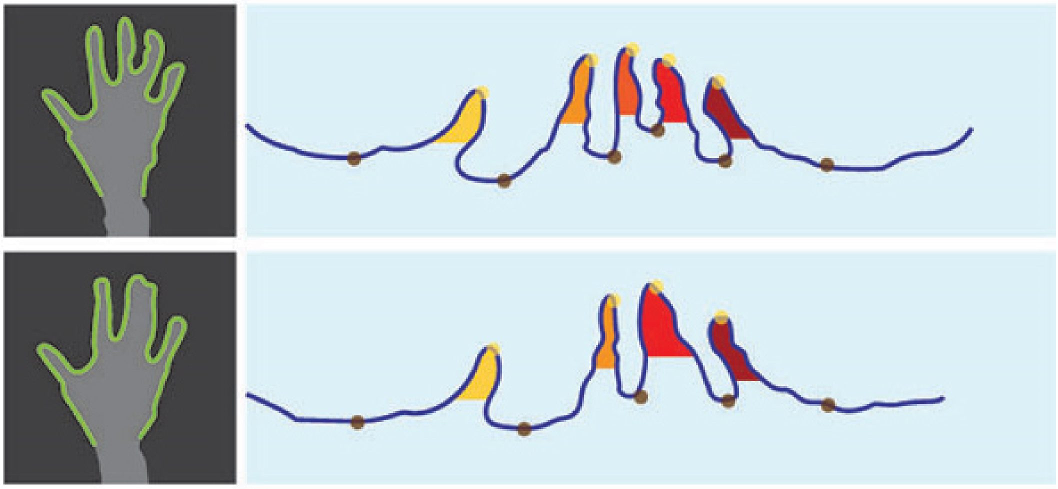


Fig. 4.7 – Curve representation

Next step of algorithm is the curve analysis with aim to find local maximum. After the curve analysis is finished, finger extraction starts. The distance of each maximum is compared to set threshold. Each finger has a specific weight and based on the defined relations fingers (relation is comparing weight with given values and based on this the number of fingers is known. i.e. if weight < 1,5*x* mean weight – segment contains one finger, etc.) are detected. This algorithm works also for joined fingers (Fig. 4.8).

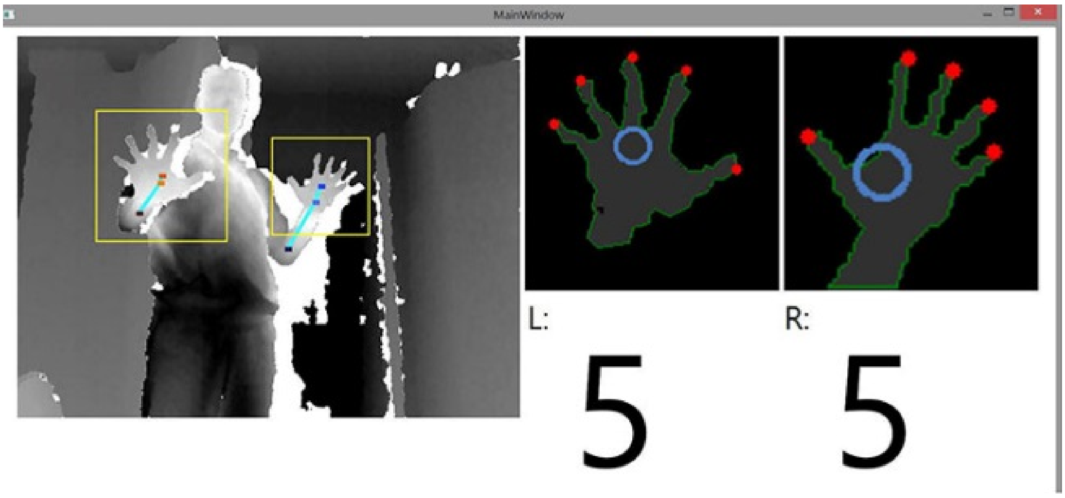


Fig. 4.8 –Detected fingers

Techniques used for dynamic gestures recognition

Dynamic gestures ensures an access to a private users' content or to control of a system or an application. Their usage can be used also as a password key. Neural networks and genetic algorithms were mostly used in the beginnings of gesture recognition.

These methods had acceptable recognition rate, but the greatest drawback was the amount of the necessary computing power and time needed for training the neural networks which were significantly and unacceptably high for practical applications.

Nowadays, new techniques are used to recognize gestures. The algorithms which do not require neuron networks have been invented, for example Golden Section Search, Incremental Recognition Algorithm and probabilistic models like Hidden Markov Model. To increase the success rate of mentioned algorithms Machine learning can be used. It exists many approaches how to do gesture recognition. HMM methods are very popular in last years. The main reason is that HMM approach is well known and used in many areas.

The algorithm proposed by Kristensson and Denby [4] originally for digital pen strokes and touch-screen devices can be extended also for dynamic gestures.

For this approach, a template is defined as a set of segments describing the gesture. It is obvious that given a set of gestures which are sufficiently distinguishable from each other the recognition may be successful after only a part of the gesture was performed (Fig. 4.9).

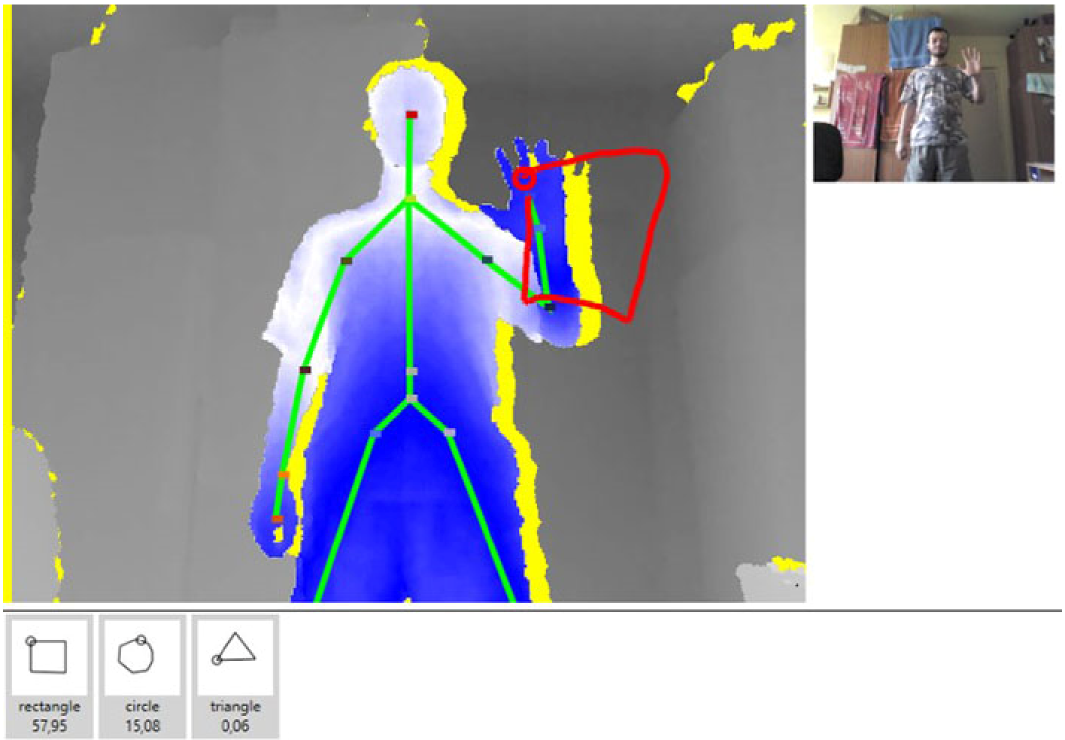


Fig. 4.9 – Preview of the test application. The red line represents the performed gesture. The templates below the picture represent the most probable gestures performed (from left to right).

The usage of gestures is extended by swipe gestures. This type of gesture brings very natural and comfortable approach. Swipe gestures are designed for fast and routine browsing in menu, programs, and gallery and contains 4 directions for each hand and couple of combination left-right hand. Method called Circle Dynamic Gesture Recognition (CDGR) published in [1] is based on hand detection, speed of movement and distance. If human hand executes a faster motion and inner circle leaves outer circle, system processes this motion and determines a gesture. The gesture is given by angle of executed motion from middle to outer circle. The possible gestures are: swipe left, right, up and down. The same gestures can also be made with both hands. User can also perform a zoom in and zoom out gestures (Fig. 4.10).

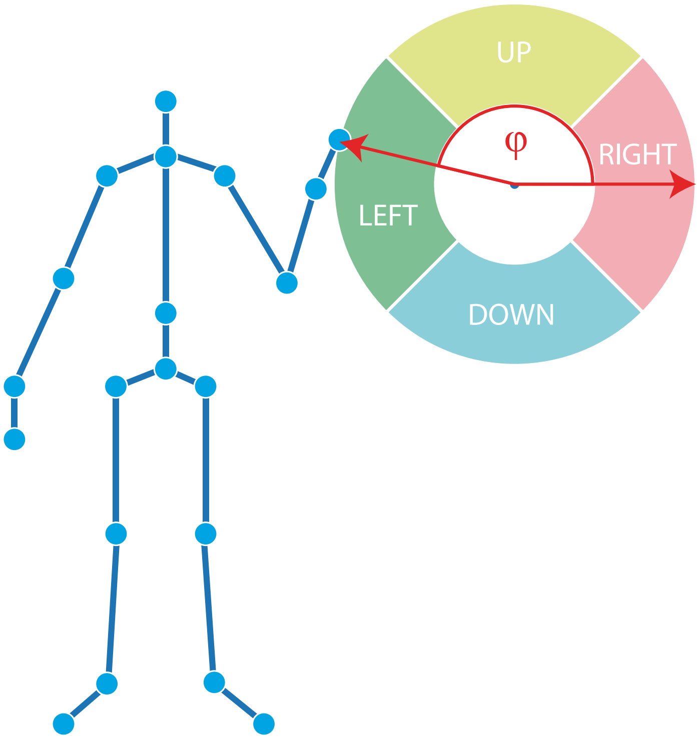


Fig. 4.10 – CDGR (circle dynamic gesture recognition)

4.5 Gestures in practice

Gestures recognition can be interpreted as one of the first possible ways how computers can understand humans or human body language. It can led to the big step when the input devices as keyboard and mouse become minority. Using the concept of gesture recognition, it is possible to point a finger at the computer screen so that the cursor will move accordingly.

Gesture navigation become more and more popular mainly in smart phones, tablets and modern TVs. Everybody knows swipe gesture to scroll up/down or move left/right. There are already applications offering set of gestures for navigation in multi touch devices, like volume control, lock the screen, control music player, take screen shot, navigate to home, back, recent-apps and menu.

The area for gestures is not limited, as example can be mentiond a tennis training application using 3D gesture recognition published by Cristian García Bauza and his team.

As an example, Samsung introduced gesture recognition in their flagship TV back in 2013. It consisted of hand tracking algorithm with recognition of the “click” gesture performed by closing the palm. In order for user to switch channels or adjust volume, he/she had to first present a raised hand to the TV, then track the shown cursor (a remnant of the computer era) to either side of the screen where buttons appeared. Then, to i.e. increase volume, user had to “click” with his/her hand several times to achieve the desired volume while keeping it in position. This approach could be considered intuitive, at least by computer-savy users, but it could hardly serve as an example of natural gesture.

5 GUI design

The *graphical user interface* (**GUI**) represents a way for interaction between humans and machines. The user interface includes hardware (physical) and software (logical) components.

The user sends command or task to the machine/system and the machine responds back by fulfilling the required command or task. In the past GUI accept input via devices such as a computer keyboard and mouse. As touchscreen devices appears, GUI adapted and also input by touch of fingers or a stylus is accepted.

The task to create suitable GUI for multimodal (or any other) application require good practice, experience and user's feedback (nowadays is this research area called user experience). GUI for multimodal applications should allow to easily, efficiently and user friendly operate a machine. It is desired the user needs to provide minimal effort to set input and also that the machine requires the minimum operational time to provide output. The connected topic to *user experience* (**UX**) is User experience design (*User experience design* **UXD**).

6 MMI applications

Modern application research is focused on natural multimodal interface and its integration into multimedia system used on daily base. Usage of multimodal interface is not limited only to the TV system but has many different applications. The vision of the future applications and system is not only to control the TV and access multimedia content, but to build system that is aware of its user's habits, preferences and interests. Such system can become an intelligent concierge of the household, and can provide advanced interconnections between various services (shopping while watching, voting or digital doorman).

In order to make the best use of multimodal interface, it's important to know which modality should be used to perform every action. Some actions will always be better executed by using a different modality. I.e. entering text would be difficult, time consuming and by all means uncomfortable using gestures, but can be easily and faster performed with voice recognition. With this in mind, it becomes necessary to introduce an integration platform that will provide multimodal applications with requested inputs. The application does not need to know the source modality, if not required explicitly.

Simple example of possible scenario: The MMI application for TV control shows a list of users identified by speech or face recognition modules. Only users in this list are permitted to control the TV using predefined set of gestures, voice commands or other modality. When the user leaves the room, he is automatically removed from the list. The other part of application shows recommended channels. Depending on user viewing preferences, system shows recommendations that best suit all users in front of the TV. Using swipe gestures a user is able to browse in this list, play or stop the video. Some of the recommended channels are locked. It means, that users without permission are not allowed to watch such content until they enter the secret pattern. For entering the secret pattern we apply dynamic gestures.