# SOM-Based Dynamic Image Segmentation for Sign Language Training Simulator

Oles Hodych[1], Kostiantyn Hushchyn[1], Yuri Shcherbyna[1], Iouri Nikolski[2], and Volodymyr Pasichnyk[2]

[1] Ivan Franko Lviv National University, Ukraine
oles.hodych@gmail.com
[2] National University "Lvivska Politechnica", Ukraine
y_nikol@yahoo.com

**Summary.** The paper discusses an image segmentation algorithm based on Self-Organising Maps and its application for the improvement of hand recognition in a video sequence. The presented results were obtained as part of a larger project, which has an objective to build a training simulator for Ukrainian Sign Language. A particular emphasis in this research is made on the image preparation for Self-Organising Map training process for the purpose of successful recognition of image segments.

**Keywords:** image segmentation, self-organising maps.

## 1 Introduction

Sign languages are based on hand signs, lip patterns and body language instead of sounds to convey meaning. The development of sign languages is generally associated with deaf communities, which may include hearing or speech impaired individuals, their families and interpreters. The only currently viable ways to enable communication between hearing impaired and not impaired people is to use services provided by interpreters (specially trained individuals or software applications), or to learn a sing language.

A manual communication has developed in situations where speech is not practical. For instance, scuba diving or loud work places such as stock exchange. In such cases learning a sign language is the most effective solution.

There is a number of commercial software packages and research projects directed at developing software applications for sign language interpretation. The majority of such software is directed at interpreting a spoken to sign language, which covers all major cases where interpretation is required for deaf people (e.g. conventions, television). For example, researchers at IBM have developed a prototype system, called SiSi (say it, sign it), which offers some novel approaches by utilising avatars to communicate speech into sign language [16]. Some other applications, such as iCommunicator [21], are based on a database of sign language videos and provide means to adaptively adjust to user's speech.

Authors of this paper are conducting a research for the purpose of creating an adaptive sign language training simulator. A large part of the core ideas and technologies

J. Yang et al. (Eds.): UNISCON 2009, LNBIP 20, pp. 29–40, 2009.

produced is already discussed in a number of articles as well as demonstrated at CeBIT 2006, 2007 and 2008. The main motivation behind this research is to provide an affordable solution for people to use the end product for self-training of the Ukrainian sign language.

One of the core ideas of this project is to use a motion camera (such as a web camera) for capturing user's hand gestures, recognise them and match against existing samples for providing trainee with a feedback as to how well the gesture was performed. When successfully implemented this approach should provide users with a self-training easy to set up environment requiring no coaching by a human trainer.

The detailed discussion of the project's progress and its key results are covered in several articles [3] [4] [5]. Gesture is a typical element of a sign language. One of the most difficult tasks, which were encountered during the research, was the requirement to recognise a sign from a sequence of video frames regardless of the background on which the gesture was performed. Our tests reveal that the adaptive technique employed for recognising user's hand (and thus the gesture) would provide a much higher success rate when performing recognition on frames with the background used for its training. The proposed training simulator should cater for a wide range of situations including different backgrounds. For example, the user of the system might be wearing a stripy shirt one day and a plain white t-shirt some other time. Thus, there was formulated a requirement to filter out the background in frames in a video sequence before attempting to recognise the gesture. During the research we've come to realize that instead of trying to identify what is the background, it is potentially more efficient to identify the smallest possible area on the image (frame), which contains the hand. The identified area could then be used for further processing to actually identify the gesture using one of the proposed earlier methods utilising the dactyl matching [5]. The process of partitioning a digital image into multiple regions is known as image segmentation. Image segmentation is typically used for locating objects or boundaries in the image, which is what's needed to locate a hand in our case.

## 2 Related Work

There are several ways to approach the problem of image segmentation. One possible way, which is more inline with our research, is to treat image segmentation as a clustering task, where objects for grouping are image pixels, and the actual groups (or clusters) are image segments. Hence the term *image clustering*, which refers to means for high-level description of the image content. Self-Organising Map and its variations were the subject of our research for the past several years [10] [11] [12], and therefore it was a natural choice for the task of image clustering.

Image segmentation is a popular research subject and there is a large number of related research projects as well as readily available commercial and open source tools. In this section we present a short overview of the published research results dedicated specifically to the use of Self-Organising Maps (SOM) as a technology for image clustering.

In [17] authors proposed the use of the two-stage process based on SOM with one-dimensional lattice. The SOM network is trained during the first stage, and in the second stage it is clustered using K-means algorithm in order to determine the number of image segments. One of the main disadvantages of the proposed technique is the use of the reduced colour information where only hue and saturation were used for preparing the data source (luminance component was excluded). As discussed later, the colour space plays a significant role in preparing the data for image clustering.

In [6] and [7] researchers present a unique data preparation scheme where not only the colour, but also the image texture was used, which proved to yield a success rate of 61.3% (with texture) comparing to 53.6% while using only colour information. Provided tests included very complex outdoor images.

A large number of human image processing approaches utilise skin detection as a key element for feature extraction. The histograms and Gaussian mixture models are amongst the most popular colour modeling methods. However, these techniques are not always best suited in the real life dynamic environments. In [2] authors proposed a SOM-based algorithm for skin detection. The accuracy of 94% was claimed on facial images.

In [1] a multi-stage clustering algorithm was proposed in application to colour image segmentation. The first stage of the proposed algorithm utilises SOM due to its distinct features of reducing the sample size and at the same time preserve the input distribution. The subsequent stages could utilise segmentation techniques that would not be possible on samples of a large size. The RGB colour space was used in this research.

When processing an image for a segmentation task it should be presented in a suitable for the segmentation algorithm way. Very often a three dimensional RGB space is used, where each or group of pixels on the image is represented as a three dimension vector. In [18] authors used both colour and spatial information. Thus utilising five dimensional vectors $(X, Y, R, G, B)$ for representing image data used in SOM training. In addition, a merging algorithm was introduced for clustered blocks to be combined into a specified number of regions with some semantic means.

Authors of [20] concentrated their research on the use of one-dimension SOM network for image segmentation. According to this research the best results were obtained for SOM with lattice configuration where the first and the last neurons are linked forming a circular structure.

Paper [22] discusses the use of an adaptive SOM colour segmentation algorithm. Similarly to [20] one dimensional SOM was used, but in this case it supported growing – pruning and merging facilities were developed to find an appropriate number of clusters automatically. Additionally, in order to further improve results in light and background changing conditions, authors developed, what they call, a *transductive* algorithm to learn the dynamic colour distribution in HSI colour space by combining supervised and unsupervised learning paradigms. The presented results yield an excellent performance. However, most of the presented tests utilised images either of a small size, thus automatically reducing the complexity, or those with a close to uniform background.

The main objective of the proposed in this paper approach is to identify a segment containing human hand in a video sequence with a non-uniform background in

a timely from the processing speed perspective manner. The following sections discuss the three key aspects of the proposed approach – selection of a colour space for image representation, data reduction for SOM training, and an information generalisation based on one video frame for segmentation of a complete sequence.

## 3 Dynamic Image Clustering

The discussion of the theoretical and algorithmic foundation of the conducted research is presented in two sections. The first sections addresses the data preparation. Specifically, the development of the most adequate method for transforming images from a video sequence into a vector space suitable for SOM consumption (i.e. training and interpretation). This phase directly effects the quality of image clustering.

### 3.1 Data Preparation

The first stage of data preparation is to choose a vector space for representing each pixel on the image. The processing speed is of high importance for fulfilling the requirement of real-time processing. The training of SOM is the most time consuming operation. Therefore, the second stage of data preparation addresses the reduction of the number of data samples used for training.

**Colour space.** It is well known that SOM operates based on the principles of the brain. One of the key SOM features is preservation of the topological order of the input space during the learning process. Some relatively recent research of the human brain has revealed that the response signals are obtained in the same topological order on the cortex in which they were received at the sensory organs [19]. One of such sensory organs are eyes, thus making the choice of SOM for analysing the visual information one of the most natural.

Colour is the brain's reaction to specific visual stimulus. Therefore, in order to train SOM for it to reflect the topological order of the image perceived by a human eye, it is necessary to choose the colour space, which closely models the way sensors obtain the visual information. The eye's retina samples colours using only three broad bands, which roughly correspond to red, green and blue light [8]. These signals are combined by the brain providing several different colour sensations, which are defined by the CIE (Commission Internationale de l'Eclairage (French), International Commission on Illumination) [15]: Brightness, Hue and Colourfulness. The CIE commission defined a system, which classifies colour according to the human visual system, forming the trichromatic theory describing the way red, green and blue lights can match any visible colour based on the eye's use of three colour sensors.

The colour space is the method, which defines how colour can be specified, created and visualised. As can be deduced from the above, most colour spaces are three-dimensional. There are more than one colour space, some of which are more suitable for certain applications than others. Some colour spaces are perceptually linear, which means that an $n$-unit change in stimulus results in the same change in perception no

matter where in the space this change is applied [8]. The feature of linear perception allows the colour space to closely model the human visual system. Unfortunately, the most popular colour spaces currently used in image formats are perceptually nonlinear. For example, BMP and PNG utilise RGB[1] colour space, JPEG utilises YCbCr, which is a transformation from RGB, HSL[2] is another popular space, which is also based on RGB.

The CIE based colour spaces, such as CIELuv and CIELab, are nearly perceptually linear [8], and thus are more suitable for the use with SOM. The CIEXYZ space devises a device-independent colour space, where each visible colour has nonnegative coordinates X, Y and Z [14]. The CIELab is a nonlinear transformation of XYZ onto coordinates $L*, a*, b*$ [14].

The image format used in our research is uncompressed 24-bit BMP (8 bit per channel), which utilises the RGB colour space. In order to convert vectors $(r, g, b) \in RGB$ into $(L*, a*, b*) \in CIELab$ it is necessary to follow an intermediate transformation via the CIE XYZ colour space. These transformations are described in details in [13] and [14]. Application of the two-step transformation to each pixel of the original image in RGB space produces a transformed image in CIELab space used for further processing.

It is important to note that when using SOM it is common to utilise Euclidean metric for calculation of distances during the learning process [19][3]. Conveniently, in CIELab space the colour difference is defined as Euclidean distance [14].

In order to demonstrate significance of the colour space selection please consider images depicted on Fig. 1.

The original image was $800 \times 600$ (refer Fig. 1(a)). Three datasets were composed based on this image using three different colour spaces: RGB, HSL and CIELab. SOM then was used to cluster these datasets. The result of this clustering is depicted on Fig. 1(b) for RGB-based dataset, Fig. 1(c) for HSL-based dataset, and Fig. 1(d) for CIELab-based dataset. As can be easily observed the CIELab-based image representation provides the best result.

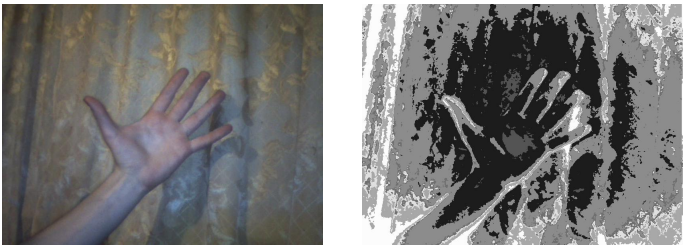
**Training dataset composition.** Instead of using every image pixel for the SOM training process, the following approach was employed to reduce the number of data samples in the training dataset.

The basic idea is to split an image into equal segments $n \times n$ pixels. Then for each such segment find two the most diverged pixels and add them to the training dataset. Finding the two most diverged pixels is done in terms of the distance applicable to the colour space used for image representation. Due to the fact that each pixel is a three dimensional vector, each segment is a matrix of vector values. For example, below is an image $A$ of $4 \times 4$ pixels in size represented in the CIELab space, and split into four segments $2 \times 2$ pixels each.

[1] Uncompressed BMP files, and many other bitmap file formats, utilise a colour depth of 1, 4, 8, 16, 24, or 32 bits for storing image pixels.

[2] Alternative names include HSI, HSV, HCI, HVC, TSD etc. [8]

[3] The selection of the distance formula depends on the properties of the input space, and the use of Euclidean metric is not mandatory.



(a) Original image (b) RGB-based image

(c) HSL-based image (d) CIELab-based image

**Fig. 1.** Image SOM processing for different colour spaces

$$A = \left(\begin{array}{cc|cc} (L_1^1, a_1^1, b_1^1)^T & (L_2^1, a_2^1, b_2^1)^T & (L_3^1, a_3^1, b_3^1)^T & (L_4^1, a_4^1, b_4^1)^T \\ (L_1^2, a_1^2, b_1^2)^T & (L_2^2, a_2^2, b_2^2)^T & (L_3^2, a_3^2, b_3^2)^T & (L_4^2, a_4^2, b_4^2)^T \\ \hline (L_1^3, a_1^3, b_1^3)^T & (L_2^3, a_2^3, b_2^3)^T & (L_3^3, a_3^3, b_3^3)^T & (L_4^3, a_4^3, b_4^3)^T \\ (L_1^4, a_1^4, b_1^4)^T & (L_2^4, a_2^4, b_2^4)^T & (L_3^4, a_3^4, b_3^4)^T & (L_4^4, a_4^4, b_4^4)^T \end{array}\right)$$

Thus, the first segment is:

$$S_1 = \left(\begin{array}{cc} (L_1^1, a_1^1, b_1^1)^T & (L_2^1, a_2^1, b_2^1)^T \\ (L_1^2, a_1^2, b_1^2)^T & (L_2^2, a_2^2, b_2^2)^T \end{array}\right)$$

The above approach can be summarised as the following algorithm. Let $n$ denote the size of segments used for image splitting, the value of which is assigned based on the image size. $T$ – the training set, which is populated with data by the algorithm. Let's also denote $j$th pixel in segment $S_i$ as $S_i(j)$. Further in the text both terms *pixel* and *vector* are used interchangeably.

The above algorithm provides a way to reduce the training dataset. It is important to note that an excessive reduction could cause omission of significant pixels resulting in poor training. At this stage it is difficult to state what rule can be used to deduce the optimal segment size. The segmentation used for the presented results was obtained though experimentation. However, even applying segmentation $2 \times 2$ pixels to an image of $800 \times 600$ pixels in size reduces the training dataset from $460000$ down to $240000$

**Algorithm 1.** Training dataset composition

*Initialisation.* Split image into segments of $n \times n$ pixels; $N > 0$ – number of segments; $T \leftarrow \varnothing$; $i \leftarrow 1$.

1. Find two the most diverged pixels $p' \in S_i$ and $p'' \in S_i$ using Euclidian distance.
   1.1 $max \leftarrow -\infty$, $j \leftarrow 1$
   1.2 $k \leftarrow j + 1$
   1.3 Calculate distance between pixels $S_i(j)$ and $S_i(k)$: $dist \leftarrow \|S_i(j) - S_i(k)\|$
   1.4 If $dist > max$ then $p' \leftarrow S_i(j)$, $p'' \leftarrow S_i(k)$ and $max \leftarrow dist$
   1.5 If $k < n \times n$ then $k \leftarrow k + 1$ and return to step 1.3
   1.6 If $j < n \times n - 1$ then $j \leftarrow j + 1$ and return to step 1.2
2. Add $p' \in S_i$ and $p'' \in S_i$ to the training set: $T \leftarrow T \cup \{p', p''\}$
3. Move to the next segment $i \leftarrow i + 1$. If $i \leq N$ then return to step 1, otherwise stop.

elements, which in turn enables the use of a smaller lattice and reduces the processing time required for SOM training.

### 3.2 Interpretation of Clusters

There are several aspects to a successful application of SOM, among which are:

- Self-organisation process, which encompasses a problem of selecting a learning rate and a neighbourhood function.
- The size and structure of the SOM lattice.

In this research the guidelines from [19] and [11] were followed to conduct the self-organisation process. The structure of the SOM lattice may differ in its dimensionality and neighbourhood relation between neurons. The use of 2-dimensional lattice with hexagonal neighbourhood relation proved to be the most efficient in our research producing more adequate clustering results comparing to other evaluated configurations.

Once the SOM structure and parameters for self-organisation process are selected, the SOM is trained on the training set $T$, which is composed for the image to be clustered. The trained SOM is then used for the actual image clustering.

As has been mentioned in previous sections, one of the most important features of SOM is topology preservation. This feature is fundamental to the proposed image segmentation approach. The basic underlying principles of which are:

- Image pixels represented by topologically close neurons should belong to the same cluster and therefore segment.
- The colour or marker used for segment representation is irrelevant as long as each segment is associated with a different one.

These two principles suggest that the position of neurons in the lattice (i.e. coordinates on the 2D plane) can be used for assigning a marker to a segment represented by any particular neuron instead of the neurons' weight vectors. This way weight vectors are used purely as references from 2D lattice space into 3D colour space, and neural

locations represent the image colour distribution. As the result of a series of conducted experiments the following formulae for calculating an RGB colour marker for each neuron have produced good results.

$$R_j \leftarrow x_j + y_j \times \lambda; G_j \leftarrow x_j + y_j \times \lambda; B_j \leftarrow x_j + y_j \times \lambda; \quad (1)$$

In formula (1) values $x_j$ and $y_j$ are coordinates of neuron $j = \overline{1, M}$, where $M$ is the number of neurons in SOM. Constant $\lambda$ should be greater or equal to the diagonal of the SOM lattice. For example, if SOM lattice has a rectangular shape of $16 \times 16$ neurons then $\lambda$ could be set to 16. Applying the same formula for R, G, and B components produces a set of gray scale colours. However, each neuron has its own colour, and one of the currently not fully resolved issues is how to group neurons based on the assigned colours into larger segments. There are several approaches, which are being currently developed to provide automatic grouping of SOM neurons into clusters [12]. However, the presented in this paper results were obtained by applying a threshold to the segmented with SOM image, which requires human interaction in specifying the threshold value. The presented image segmentation approach can be summarised as the following algorithm.

**Algorithm 2.** Image segmentation

*Initialisation.* $p_j = (R_j, G_j, B_j)$ – pixel $j$; $j = \overline{1, K}$; $K > 0$ – total number of pixels; $j \leftarrow 1$; $i^*(p_j) = (R_{i^*}, G_{i^*}, B_{i^*})$ – a weight vector of the best matching unit (BMU – winning neuron) for input vector $p_j$; $(x_{i^*}, y_{i^*})$ – coordinates of neuron $i^*$; choose appropriate values for $\lambda$.

1. Find $BMU(p_j)$ for vector $p_j$ in the trained SOM utilising the distance used for training (Euclidian for CIELab).
2. Calculate marker for pixel $p_j$: $R_j \leftarrow x_{i^*} + y_{i^*} \times \lambda$, $G_j \leftarrow R_j$, $B_j \leftarrow R_j$.
3. Move to the next image pixel: $j \leftarrow j + 1$;
4. If $j \leq K$ return to step 1, otherwise stop.

## 4 Experimental Results

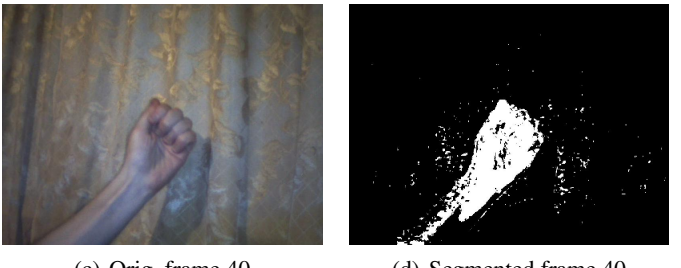
In this section we would like to demonstrate some results of this research. A special interest is the case of training SOM on one of the frames in a video sequence and using it for segmentation of subsequent frames. The following figures present results by depicting the original and segmented images, which correspond to frames number 25 through to 60 with step of 5 frames of the recorded video. The training of SOM and determining of the appropriate threshold value was performed only on the first image (i.e. 25th frame). The recorded video captured an open palm closing and opening again during a period of several seconds. The recording was done using an ordinary PC web camera capable of 30FPS throughput with a frame size of $800 \times 600$ pixels. The background of the captured scene is nonuniform, which increases the complexity of image segmentation.

(a) Orig. frame 25 (b) Segmented frame 25

(c) Orig. frame 30 (d) Segmented frame 30

**Fig. 2.** Frame 25 and 30

(a) Orig. frame 35 (b) Segmented frame 35



(c) Orig. frame 40 (d) Segmented frame 40

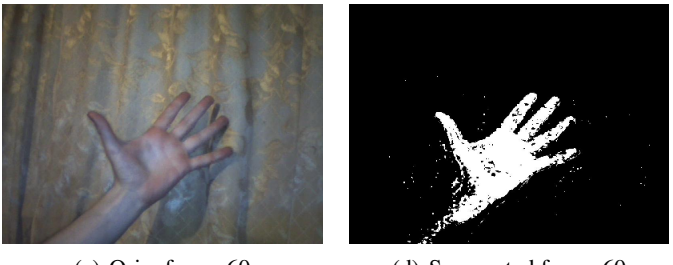
**Fig. 3.** Frame 35 and 40

(a) Orig. frame 45 (b) Segmented frame 45

(c) Orig. frame 50 (d) Segmented frame 50

**Fig. 4.** Frame 45 and 50

(a) Orig. frame 55 (b) Segmented frame 55



(c) Orig. frame 60 (d) Segmented frame 60

**Fig. 5.** Frame 55 and 60

Figure 2 depicts a fully open palm corresponding to frame 25, which was used for SOM training, and frame 30 depicting a palm with slightly contacted fingers.

Frames 35 and 40 correspond to palm closing, which are depicted in figures 3. As can be observed, the contracted palm is successfully separated from the background by the proposed segmentation algorithm. However, at the same time a slightly greater number of artifacts, which do not belong to the palm, are captured.

Figures 4 depict frames 45 and 50 where palm started opening again.

The remaining figure 5 depicts the final frames that captured the full opening of the palm. These frames are very similar to frames 25 and 30, therefore, the good segmentation results where expected.

The key aspect of the presented in this section results is the use of SOM trained only on a single frame. This initial frame as well as all subsequent ones have been successfully segmented with clear separation of the human palm from the nonuniform background. Although, some elements of the background were recognised as part of the same segment and caused minor undesired artifacts scattered around the palm. The use of only one frame for SOM training provides a provision for much faster dynamic image segmentation needed for video, avoiding SOM retraining for every frame.

## 5 Conclusion and Future Work

The main purpose of developing an image segmentation algorithm in our case is to improve image analysis for dactyl matching. The proposed approach showed good results not only for human hand recognition, and potentially can be used for other applications. The main disadvantage of the developed approach is the need for human interaction when specifying the threshold values in the final step of image segmentation. However, this aspect is being currently addressed utilising the results obtain in [12], which allows automatic clustering of the trained SOM. Another important subject of the future research direction is increasing the quality of segmentation by applying hierarchical clustering. The basic idea behind this approach is to start SOM training on images with reduced information, following additional training based on the same image with increased information. There are many image smoothing methods that provide a way of controlling the amount of image details, which may impact the quality of information reduction and thus segmentation results.

## References

1. Akgül, C.B.: Cascaded self-organizing networks for color image segmentation (2004), http://www.tsi.enst.fr/~akgul/oldprojects/CascadedSOM_cba.pdf
2. Brown, D., Craw, I., Lewthwaite, J.: A SOM Based Approach to Skin Detection with Application in Real Time Systems, University of Aberdeen (2001), http://www.bmva.ac.uk/bmvc/2001/papers/33/accepted_33.pdf
3. Davydov, M.V., Nikolskyi, Y.V.: Automatic identification of sign language gestures by means on dactyl matching. Herald of National University "Lvivska Polytechnica" 589, 174–198 (2007)
4. Davydov, M.V., Nikolskyi, Y.V., Pasichnyk, V.V.: Software training simulator for sign language learning. Connection, 98–106 (2007) (in Ukrainian)

5. Davydov, M.V., Nikolskyi, Y.V., Pasichnyk, V.V.: Selection of an effective method for image processing based on dactyl matching for identification of sign language gestures. Herald of Kharkiv National University of Radio-Electronics 139, 59–68 (2008) (in Ukrainian)
6. Campbell, N.W., Thomas, B.T., Troscianko, T.: Neural Networks for the Segmentation of Outdoor Images. In: International Conference on Engineering Applications of Neural Networks, pp. 343–346 (1996)
7. Campbell, N.W., Thomas, B.T., Troscianko, T.: Segmentation of Natural Images Using Self-Organising Feature Maps, University of Bristol (1996)
8. Ford, A., Roberts, A.: Colour Space Conversions (1998), http://www.poynton.com/PDFs/coloureq.pdf
9. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, San Francisco (2001)
10. Hodych, O., Nikolskyi, Y., Shcherbyna, Y.: Application of Self-Organising Maps in medical diagnostics. Herald of National University "Lvivska Polytechnica" 464, 31–43 (2002)
11. Hodych, O., et al.: Analysis and comparison of SOM-based training algorithms. Control Systems and Machines 2, 63–80 (2006) (in Ukrainian)
12. Hodych, O., et al.: High-dimensional data structure analysis using Self-Organising Maps. In: 9th International Conference, CAD Systems in Microelectronics. CADSM apos 2007, Feburary 2007, pp. 218–221 (2007)
13. Hoffmann, G.: CIE Color Space (2000), http://www.fho-emden.de/~hoffmann/ciexyz29082000.pdf
14. Hoffmann, G.: CIELab Color Space (2003), http://www.fho-emden.de/~hoffmann/cielab03022003.pdf
15. Hunt, R.W.G.: Measuring Colour, 3rd edn. Fountain Pr Ltd. (2001)
16. IBM Research Demonstrates Innovative Speech to Sign Language Translation System, Press-release (September 12, 2007), http://www-03.ibm.com/press/us/en/pressrelease/22316.wss
17. Moreira, J., Da Fontoura Costa, L.: Neural-based color image segmentation and classification using self-organizing maps (1996), http://mirror.impa.br/sibgrapi96/trabs/pdf/a19.pdf
18. Jiang, Y., Chen, K.-J., Zhou, Z.-H.: SOM Based Image Segmentation. LNCS (LNAI), vol. 2639, pp. 640–643. Springer, Heidelberg (2003)
19. Kohonen, T.: Self-Organizing Maps, 3rd edn. Springer, Heidelberg (2001)
20. Reyes-Aldasoro, C.C.: Image Segmentation with Kohonen Neural Network Self-Organising Maps (2004), http://www.cs.jhu.edu/ cis/cista/446/papers/SegmentationWithSOM.pdf
21. The iCommunicator User's Guide (2005), http://www.myicommunicator.com/downloads/iCommunicator-UserGuide-v40.pdf
22. Wu, Y., Liu, Q., Huang, T.S.: An Adaptive Self-Organizing Color Segmentation Algorithm with Application to Robust Real-time Human Hand Localization. In: Proc. Asian Conf. on Computer Vision, Taiwan, (2000)