DETECTION OF SIGN-LANGUAGE CONTENT IN VIDEO THROUGH POLAR MOTION PROFILES

Virendra Karappa, Caio D. D. Montero, Frank M. Shipman, and Ricardo Gutierrez-Osuna
Department of Computer Science and Engineering, Texas A&M University
{vk2382,shipman,rgutier}@cse.tamu.edu, caioduarte.diniz@gmail.com

ABSTRACT

Locating sign language (SL) videos on video sharing sites (e.g., YouTube) is challenging because search engines generally do not use the visual content of videos for indexing. Instead, indexing is done solely based on textual content (e.g., title, description, metadata). As a result, untagged SL videos do not appear in the search results. In this paper, we present and evaluate a classification approach to detect SL videos based on their visual content. The approach uses an ensemble of Haar-based face detectors to define regions of interest (ROI), and a background model to segment movements in the ROI. The two-dimensional (2D) distribution of foreground pixels in the ROI is then reduced to two 1D polar motion profiles by means of a polar-coordinate transformation, and then classified by means of an SVM. When evaluated on a dataset of user-contributed YouTube videos, the approach achieves 81% precision and 94% recall.

Index terms– content-based video retrieval, sign language, metadata extraction

1 INTRODUCTION

Sign Languages (SL) rely on hand gestures combined with facial expressions and postures of the body to convey their message. It is the primary medium of communication for many who are deaf and hard-of-hearing [1], and serves as a substitute for spoken communication. Because sign language is a visual form of communication, video sharing websites can be very beneficial to the deaf community as a means to exchange information.

Even though the number of SL videos uploaded to the web is increasing rapidly, only a small subset of these videos are easily available to the deaf community. The main reason for this mismatch is that search engines index videos based only on their associated text descriptions or tags. However, for many SL videos the text descriptions are associated with the topic (e.g., sports, politics) rather than the language being used (i.e., American Sign Language). Therefore, such videos do not show up in the search results when performing standard text queries with keywords such as American Sign Language, British Sign Language, etc. Given the size of user contributed video sites, manual tagging is prohibitive. Instead, meaningful improvement of search results requires automated tagging. This in turn requires algorithms to detect sign language content based on visual information alone.

In this paper, we present and evaluate a technique to detect SL content in user contributed videos. Our approach relies on face detection and background modeling techniques, combined with a polar representation of hand movements. In a first step, we detect faces in the videos using an ensemble of face detectors based on Haar-like features [2, 3]. We then extract foreground hand gestures by subtracting a background model based on adaptive Gaussian mixtures [4]. For each frame, we then calculate the proportion of foreground pixels along the two polar coordinates (angle, distance) on a reference frame centered on the signer's face and scaled to the face's proportions –to provide translation and scale invariance. These polar motion profiles (PMPs) capture the amount of signing activity in each frame of the video. We average PMPs across all the frames to obtain a single PMP for each video. Evaluation of this PMP representation on a collection of YouTube videos shows that an SVM classifier can achieve 81% precision and 94% recall.

Relation to prior work. The work presented here grows out of a pilot study published at ASSETS 2012 [5]. The objective of that pilot study was to establish proof-of-concept for the feasibility of detecting SL in videos. For that reason, that study was performed on a constrained video dataset with videos containing a single signer and a static background, with movements being mainly those of the signer. This allowed us to use a low-pass filter as a background detection model and a simple feature-extraction technique that computed the amount and symmetry of movements around the face. The work presented here relaxes those assumptions by considering videos that contain multiple signers and complex non-stationary backgrounds. This required more robust techniques for face detection and background modeling, as well as a richer feature representation of hand movements.

The rest of the paper is organized as follows. Section 2 provides a summary of past work on sign language recognition and its applicability to our study. Section 3 describes the proposed video processing and feature extraction approach for detecting SL content. Section 4 describes the video collections used for validation and classification results. The paper concludes with a discussion and directions for future work.

2 BACKGROUND / RELATED WORK

The majority of the work on video-based SL detection has focused on transcription (i.e., recognizing the specific signs being made). In one of the earliest studies, Starner et al [6] developed an HMM classifier capable of recognizing 40 American Sign Language (ASL) words for a single signer. Later work by Vogler and Metaxas [7] showed that parallel HMMs can be used to scale the vocabulary size; however, the parallel HMMs was tested on a vocabulary of only 22 signs. Yang et al. [8] used a vocabulary size of 40 ASL signs to extract motion trajectories. They trained a time-delay-neural-network using these trajectories and achieved 93% recognition on unseen test trajectories. By incorporating motion information, Parashar et al. [9] represented 39 signs using relational distributions of hand motions to construct a space of probability functions.

Thus far, recognition of larger vocabularies has been possible only with data gloves. As an example, Liang et al. [10] used data gloves to recognize Taiwan sign language gestures with a vocabulary of between 71 and 250 words. Their system required that the gestures were performed slowly in order to detect word boundaries. Along the same lines, Fang et al. used 18-sensor data-gloves and three position trackers to extract hand motion information [11].

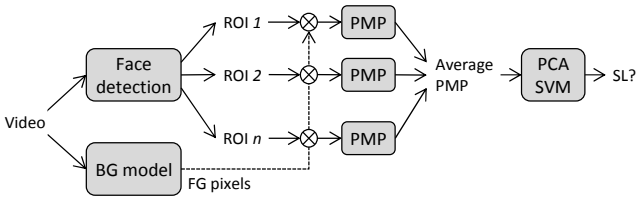

Fig. 1. Signal processing pipeline. ROI: region of interest; BG/FG: background/ foreground; PMP: polar motion profile

Other approaches involve image matching and hand shape recognition techniques. Somers and Whyte [12] built 3D models and silhouettes of 36 hand shapes to recognize Irish sign language. Dimov et al. [13] treated sign language recognition as a content-based image retrieval problem by searching the input sign image in a database of signs in the form of static images. In another data-driven approach, Potamias et al. [14] used nearest neighbor search of images for hand shape recognition.

Approaches developed for sign language transcription are of limited value in our context in that most of them work only modestly with relatively small vocabularies, or are signer-dependent and require large amounts of training data. Fortunately, detecting sign language is a much simpler problem than translating it. As an example, Cherniavsky et al. [15] developed an activity detection technique for cell-phone cameras that could determine whether a user was signing or not with 91% accuracy, even in the presence of noisy (i.e., moving) backgrounds. The algorithm was used to determine when the video phone user was signing and when they were watching the video of their conversational partner in order to effectively use network bandwidth during a sign language conversation on mobile devices. Thus, it is unlikely this algorithm would be as successful in distinguishing between sign language videos and other videos involving people gesturing.

3 METHODS

The overall signal processing pipeline for our SL classification method is illustrated in Fig. 1. In a first step, we process the video with a face-detection algorithm to locate regions of interest (ROI) at each frame. In parallel, we generate a background model for each video, from which we identify foreground objects at each frame. At each ROI, we then extract a polar motion profile (PMP) that represents the probability of foreground objects at each polar coordinate; see Fig. 2(f). An average PMP is computed for each video by averaging across ROIs and frames. This average PMP is finally passed to an SVM classifier to determine whether the video contained SL content. Details on each of these steps are provided on the following subsections.

3.1 Face detection

In our previous work [5], we used a single Haar-cascade classifier for face detection [2]. This method works well with static backgrounds, but does not generalize with videos that contain dynamic backgrounds. Further, the classifier was constrained to searching for a single face and therefore failed when multiple signers were present in a frame. To address these issues, the new algorithm uses multiple Haar-cascade recognizers in parallel[1], each cascade returning a list of bounding rectangles (one rectangle for each candidate location for a face). To remove false positives, the algorithm takes a majority vote by testing whether there are three or more overlapping rectangles at each candidate location, with the constraint that each rectangle originates from a different cascade.

Fig. 1(a-b) illustrates face-detection results on a video containing multiple faces; the individual cascades return multiple potential faces, many of which are false positives; taking a majority vote as described above eliminates all false positives and returns the location of the three faces in the frame.

3.2 Background modeling

Once (and if) a face has been detected, we apply background subtraction to extract foreground objects within an ROI surrounding the face. Following [4], we model the color distribution at each pixel in the video using a separate probability density function per pixel; this is necessary since individual pixels can have vastly different statistics across the video, particularly with non-stationary backgrounds. We build a background model for each pixel with an adaptive Gaussian mixture model (GMM) as proposed by [4]. In this approach, the background model is trained on a set of pixel values $X_T = \{x^{(t)}, \dots, x^{(t-T)}\}$ obtained for a time period T. The background model is denoted by $\hat{p}(\vec{x}/X_T, BG)$, a GMM of maximum M components as shown below

$$\hat{p}(\vec{x}/X_T, BG) = \sum_{m=1}^{M} \hat{\pi}_m N(\vec{x}; \hat{\vec{\mu}}_m, \hat{\sigma}_m^2 I)$$

where $\hat{\pi}_m$, $\hat{\vec{\mu}}_m$, and $\hat{\sigma}_m$ are the mixing weights, estimated means, and estimated variances of the m^{th} Gaussian component.

For every new data sample $x^{(t)}$ at time t, these parameters are updated using the following equations:

$$\hat{\pi}_m \leftarrow \hat{\pi}_m + \alpha\left(o_m^{(t)} - \hat{\pi}_m\right) - \alpha c_T$$
$$\hat{\vec{\mu}}_m \leftarrow \hat{\vec{\mu}}_m + o_m^{(t)}(\alpha/\hat{\pi}_m)\,\vec{\delta}_m$$
$$\hat{\sigma}_m^2 \leftarrow \hat{\sigma}_m^2 + o_m^{(t)}(\alpha/\hat{\pi}_m)\,(\vec{\delta}_m^T\vec{\delta}_m - \hat{\sigma}_m^2)$$

where $\alpha = 1/T$ describes an exponentially decaying envelope to limit the influence of old data; αc_T is a negative bias that adjusts the number of components automatically (components with negative weights are dropped); $\vec{\delta}_m$ is the Mahalanobis distance between the data sample and the m^{th} component; and $o_m^{(t)}$ represents the ownership of the t^{th} data sample $\vec{x}^{(t)}$ ($o_m^{(t)}$is set to 1 if it lies within three standard deviations, and otherwise is set to 0).

If the data sample $\vec{x}^{(t)}$ is outside three standard deviations for all the components, a new component is generated with $\hat{\pi}_{M+1} = \alpha, \hat{\vec{\mu}}_{M+1} = \vec{x}^{(t)}, \hat{\sigma}_{M+1} = \sigma_0$, where σ_0 is the initial variance (determined empirically). The component with smallest $\hat{\pi}_m$ is dropped when the maximum number of components M is reached.

Foreground objects appearing in the scene introduce data samples $\vec{x}^{(t)}$ which are not close to any of Gaussian components. New components are generated for these with smaller weights. Thus, the background model can be approximated with the first B components with largest weights:

$$\hat{p}(\vec{x}/X_T, BG) \sim \sum_{m=1}^{B} \hat{\pi}_m N(\vec{x}; \hat{\vec{\mu}}_m, \hat{\sigma}_m^2 I)$$

The components can be included in background only when sum of their weights is greater than a certain tunable threshold. If the weights are sorted by descending order, the number of largest components to be included in background is given by:

[1] These face detectors were developed by Lienhart et al. [3], [16] and are provided as part of the openCV library [17].

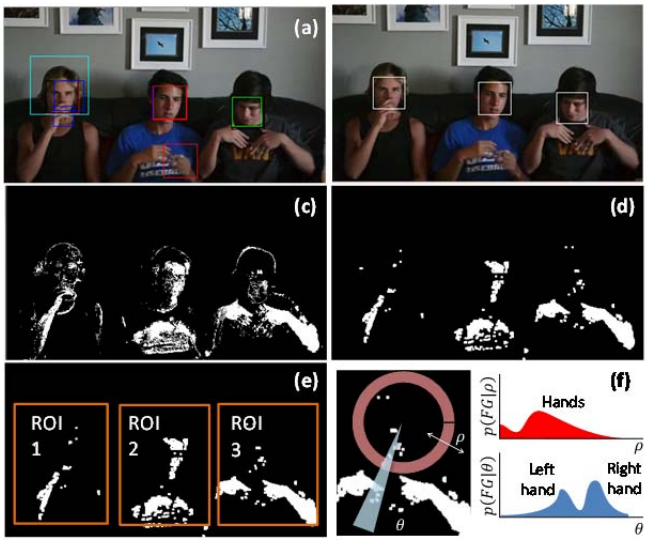

Fig. 2. (a) Faces detected by multiple Haar cascades, each denoted by a colored box (b) Post-processing of (a) by taking a majority vote. (c) Foreground (FG) pixels returned by the background model. (d) Refined FG after morphological de-noising. (e) ROIs defined for each face detected in the frame. (f) Computation of PMPs for a video frame. The PMP represent the probability of finding a FG pixel. Maxima at each PMP indicates the position of the hands on an SL video.

$$B = \arg\min_{b}\left(\sum_{m=1}^{b} \hat{\pi}_m > (1 - c_f)\right)$$

where c_f is a measure of the amount of data that should be included in the foreground. In our case, c_f was tuned empirically to optimize the detection of hand movements.

Fig. 2 (c) shows segmentation results obtained by applying the adaptive GMM background subtraction method. As a final step, we apply morphological erosion and dilation to remove small foreground objects; results are shown in Fig. 2(d). This distribution of foreground pixels (on a frame by frame basis) is then used to generate polar motion profiles, as described next.

3.3 Extraction of Polar Motion Profiles

In a final step, we combine results from the face-detection and background-segmentation algorithms to extract a representation of foreground (moving) objects around each face. For every face detected on a frame, we define a region of interest (ROI) large enough[2] to span the range of hand motions in SLs; see Fig. 2(e).

Once ROIs have been defined for each frame, we generate a polar motion profile (PMP) for each. The PMP is a translation-and-scale-invariant measure of the amount of signing activity computed on a polar coordinate system centered on each face and scaled to the dimensions of each face; see Fig. 2(f). For each ROI, it is computed as the ratio of foreground to total number of pixels at each polar co-ordinate (ρ, θ):

$$PMP_i(\theta, t) = FG_i(\theta, t)/(FG_i(\theta, t) + BG_i(\theta, t))$$

where $FG_i(\theta, t)$ denotes the number of foreground pixels at angular position θ for the i-th ROI of frame t, and $BG_i(\theta, t)$ is the corresponding number of background pixels. The overall motion profile of the video is calculated as the average PMP across ROIs and frames:

$$PMP(\theta) = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{R(t)}\sum_{r=1}^{R(t)} PMP_i(\theta, t)$$

where $R(t)$ is the number of ROIs at frame t and T is the number of frames in the video. The same process is used to derive a PMP for the second polar coordinate (ρ).

4 RESULTS

We validated the SL classification method on two datasets (A, B). **Dataset A** [5] was designed to match to the assumptions of our pilot study (static backgrounds, single signers). The dataset contained 192 videos, including 98 SL videos and 94 non-SL videos. The majority of the non-SL videos had been selected by browsing for likely false-positives based on visual analysis (e.g. the whole video consisted of a gesturing presenter, weather forecaster, or other person moving their hands and arms.) This dataset was used to compare our SL classification against the previous method [5] under ideal conditions for the latter.

Dataset B relaxed the assumptions of the pilot study. These videos were selected by performing the text query "*American Sign Language*" using YouTube's search function. We manually labeled as SL/non-SL the top 105 results returned by the search; the majority of these videos did actually contain SL, with only a few false positives (5%). To obtain a set of non-SL videos, we considered related video recommendations for the top 105 results from the search. Again, we manually labeled these related videos and selected 100 videos which did not contain SL. A majority of the videos in dataset B consisted of complex backgrounds, titles and captions appearing intermittently, and multiple signers.

4.1 Average polar motion profiles

Fig. 3 illustrates the average polar motion profiles for SL and non-SL videos on both datasets. The angular profile $PMP(\theta)$ for SL videos shows a high proportion of foreground pixels (i.e., moving objects) at angles near $\theta = 270°$, which correspond to hand positions directly below the signer's face. In contrast, non-SL videos show activity not only at $\theta = 270°$ but also at angles near $\theta = 90°$, which correspond to hand positions directly above the face. These results are consistent for both datasets (A, and B) which points to the generality of the angular profiles as a measure of discrimination between SL and non-SL videos.

The radial profile $PMP(\rho)$ for SL videos on database A show a high proportion of foreground pixels at a broad range of distances ranging from 20% to 80% of the maximum distance, relative to the size of the ROI[2]. In contrast, the distribution of foreground pixels for non-SL on database A peaks at around 30% of the maximum distance, and remains rather constant at larger distances. On dataset B, however, the radial profiles for SL and non-SL videos are very similar, which suggest that radial profiles may not be a reliable measure of discrimination between SL and non-SL videos.

4.2 Retrieval results

We compared the PMP classifier against the one in our pilot study [5], which we will refer to as 5FC (five feature classifier); see Appendix A for details on this earlier classifier.

[2] The face-detection module returns a bounding box of size $H \times W$. From this, we define an ROI to cover $1H$ above the face center, $3H$ below the face center, $2W$ to the right of the face center, and $2W$ to the left.

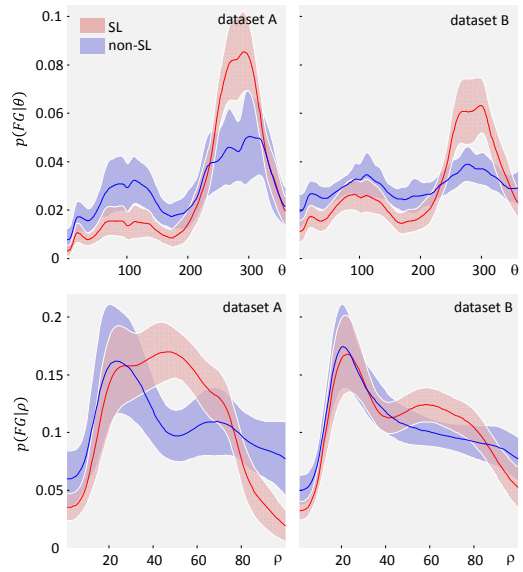

Fig. 3. Average PMPs for SL and non-SL videos on both datasets. Shaded bands represent one-half standard deviation.

Prior to classification, we used principal components analysis to reduce the dimensionality of each PMP down to five features; this ensured a fair comparison between both classifiers (5FC, PMP). We then trained an SVM classifier with an RBF kernel, using the principal components as features. Classification results on dataset A are shown in Table 1. These results represent the average of 1,000 executions, with training and test sets selected randomly each time. We compared both classifiers as a function of the training set size (15, 30, 45 and 60 videos per class) in terms of precision, recall and F1 score (harmonic mean of precision and recall). Both methods perform comparably, with a slight advantage for 5FC in terms of precision and a slight advantage for PMP in terms of recall. Comparison of the F1 scores also shows a small advantage towards PMP.

In a second experiment, we used the classifiers trained on dataset A to generate class labels for the videos in dataset B, a more challenging test since both datasets had been constructed for different purposes. Table 2 summarizes the classification results on dataset B. Both classifiers achieve similar precision rates as in dataset A. However, while PMP is able to maintain the high recall rate obtained on dataset A, recall degrades dramatically for 5FC. As a result, comparison of F1 scores reveals a strong superiority for the PMP classifier.

Table 1 Results for both classifiers (5FC, PMP) on dataset A as a function of training set size

#videos/class	Precision		Recall		F1 score	
	5FC	PMP	5FC	PMP	5FC	PMP
15	**0.82**	0.78	0.86	**0.90**	**0.84**	0.83
30	**0.84**	0.81	0.88	**0.92**	0.85	**0.86**
45	0.81	**0.82**	0.91	**0.93**	0.85	**0.87**
60	0.82	**0.82**	0.91	**0.93**	0.86	**0.87**

Table 2 Results for both classifiers on the new dataset

Classifier	Precision	Recall	F1 score
5FC	**0.82**	0.60	0.69
PMP	0.81	**0.94**	**0.87**

5 DISCUSSION

We have presented an approach to detect sign-language content in videos. The approach combines multiple face detectors and an adaptive background model to extract ROIs. From these, we generate a distribution of foreground objects in polar coordinates – the polar motion profiles (PMP). We evaluated the approach against our previous classifier (5FC) on the original dataset in [5] (dataset A), and a new dataset collected from YouTube (dataset B).

When tested on dataset A, both methods provide similar performance, an expected result since the dataset was designed to match the assumptions of 5FC (single signers, static backgrounds.) In contrast, dataset B was designed to represent typical videos returned by text queries, and includes videos with complex backgrounds, titles and captions, and multiple signers. When tested on dataset B, both methods achieve similar precision rates (81-82%). When considering recall rates, however, the classifiers perform very differently, with PMP increasing to 94% and 5FC dropping to a low 60% recall rate. This result can be attributed to several factors, including the richer representation provided by PMPs, the multiple face detection cascades that add robustness, the improved background modeling capable of handling non-stationary backgrounds, and the ability to track multiple signers.

One may be tempted to question the relevance of symmetry in hand movements as a discriminating feature for SL – a main finding of our earlier study [5]. However, close inspection of Fig. 3 reveal a strong vertical symmetry in the angular profiles, with higher probability of foreground motions around $\theta = 270°$ (directly below the signer's face). Thus, symmetry appears to be a key and robust feature in discriminating SL – the angular profiles are qualitatively similar for both datasets. In contrast, the radial profiles $PMP(\rho)$ appear to be less reproducible across datasets

Several areas for improvement are being pursued at this time. First, additional work is underway to improve recognition of profile faces – our current method uses a single Haar cascade for profile faces. Adding an adaptive skin-detection module would further improve segmentation of hand motions. Finally, our current method generates an average motion profile for each video, ignoring cues that may be present in the sequence of these motions; as an example, an HMM trained on the sequence of principal components projections may be used to extract dynamic signatures of sign languages and further improve retrieval results.

6 ACKNOWLEDGMENTS

We would like to thank Avinash Parnandi, Jin Huang, Sandesh Aryal and Chris Liberatore for their help with multiple drafts of this manuscript. This material is based upon work supported in part by NSF award DUE 09-38074, and a gift from Microsoft Corp.

7 APPENDIX A

Our prior SL detection method [5] used a single Haar-cascade for face detection, and a low-pass filter for background modeling:

$$BG(t) = (1-\alpha)\, BG(t-1) + \alpha x(t)$$

where $x(t)$ is the grayscale value of the pixel at time t, and $\alpha = 0.04$. The method then extracted five features to characterize (1) the total amount of activity per video, (2) the spread of activity across the scene, (3), the continuity of motion from frame to frame, (4) the symmetry of motions with respect to the face, and (5) the amount of non-facial movements.

8 REFERENCES

[1] NIH, "American Sign Langauge," *NIH Publication No. 11-4756*, June 2011.

[2] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," presented at the Proc. 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2001.

[3] R. Lienhart and J. Maydt, "An extended set of Haar-like features for rapid object detection," presented at the Proc. 2002 Intl. Conf. on Image Processing, 2002.

[4] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," presented at the Proc. 17th Intl. Conf. on Pattern Recognition (ICPR), 2004.

[5] C. D. Monteiro, R. Gutierrez-Osuna, and F. M. Shipman, "Design and evaluation of classifier for identifying sign language videos in video sharing sites," presented at the Proc. 14th Intl. ACM SIGACCESS Conference on Computers and Accessibility (ASSETS), 2012.

[6] T. Starner, J. Weaver, and A. Pentland, "Real-time american sign language recognition using desk and wearable computer based video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 1371-1375, 1998.

[7] C. Vogler and D. Metaxas, "Parallel hidden markov models for american sign language recognition," presented at the Proc. Seventh IEEE Int. Conf. on Computer Vision, 1999.

[8] M.-H. Yang, N. Ahuja, and M. Tabb, "Extraction of 2d motion trajectories and its application to hand gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1061-1074, 2002.

[9] A. S. Parashar, "Representation and interpretation of manual and non-manual information for automated American sign language recognition," University of South Florida, 2003.

[10] R.-H. Liang and M. Ouhyoung, "A real-time continuous gesture recognition system for sign language," presented at the Proc. Third IEEE Intl. Conf. on Automatic Face and Gesture Recognition, 1998.

[11] G. Fang, W. Gao, X. Chen, C. Wang, and J. Ma, "Signer-independent continuous sign language recognition based on SRN/HMM," in *Gesture and sign language in human-computer interaction*, ed: Springer, 2002, pp. 76-85.

[12] G. Somers and R. Whyte, "Hand posture matching for irish sign language interpretation," presented at the Proc. 1st Int. Symp. on Information and Communication Technologies, 2003.

[13] D. Dimov, A. Marinov, and N. Zlateva, "CBIR approach to the recognition of a sign language alphabet," presented at the Proc. 2007 Intl. Conf. on Computer Systems and Technologies, 2007.

[14] M. Potamias and V. Athitsos, "Nearest neighbor search methods for handshape recognition," presented at the Proc. 1st Intl. Conf. on Pervasive Technologies Related to Assistive Environments, 2008.

[15] N. Cherniavsky, R. E. Ladner, and E. A. Riskin, "Activity detection in conversational sign language video for mobile telecommunication," presented at the Proc. 8th IEEE Intl. Conf. on Automatic Face & Gesture Recognition, 2008.

[16] R. Lienhart, A. Kuranov, and V. Pisarevsky, "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection," in *Pattern Recognition*. vol. 2781, B. Michaelis and G. Krell, Eds., ed: Springer Berlin Heidelberg, 2003, pp. 297-304.

[17] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*: O'reilly, 2008.