16th European Signal Processing Conference (EUSIPCO 2008), Lausanne, Switzerland, August 25-29, 2008, copyright by EURASIP

FINGER-SPELLING RECOGNITION WITHIN A COLLABORATIVE SEGMENTATION/BEHAVIOR INFERENCE FRAMEWORK

Laura Gui[1]*, Jean-Philippe Thiran*[1]*, and Nikos Paragios*[2]

[1] IEL - STI, Ecole Polytechnique Fédérale de Lausanne
CH-1015 Lausanne, Switzerland
phone: +41 21 693 4622, fax: +41 21 693 7600, email: {laura.gui, JP.Thiran}@epfl.ch
web: ltswww.epfl.ch
[2] Laboratoire MAS, Ecole Centrale de Paris
Grande Voie des Vignes, 92 295 Chatenay-Malabry, France
phone: +33 (0)1 4113 1785, fax: +33 (0)1 4113 1735, email: nikos.paragios@ecp.fr
web: www.mas.ecp.fr

ABSTRACT

We introduce a new approach for finger-spelling recognition from video sequences, relying on the collaboration between the feature extraction and behavior inference processes. The inference process dynamically guides the segmentation-based feature extraction process towards the most likely location of the signer's hand (based on its attributes). Reciprocally, segmentation offers to the inference process hand object attributes extracted from each image, combining the received guidance and new image information. This collaboration is beneficial for both processes, yielding not only accurate segmentations of the spelling hand, but also a robust recognition scheme, which can cope with complex backgrounds, typical of real life situations.

1. INTRODUCTION

In sign languages, information is mostly conveyed through a word-level sign vocabulary, combining arm and body motions, hand shapes and facial expressions. Finger-spelling is the component which connects a sign language with the surrounding (spoken) languages. It consists of manual representations of alphabet letters and it is used for spelling words that have no sign equivalent (e.g. proper nouns or foreign words).

For word-level sign recognition, the most successful approaches [1, 2] rely on the use of devices such as data-gloves and magnetic trackers. Compared to these, purely vision-based approaches are preferable, being cheaper in terms of technology and also less cumbersome for the signer. Among these, the American Sign Language (ASL) recognizer proposed in [3] tracks hands as skin-colored blobs, extracts global features (e.g. positions and inertia axis angles), and classifies them via Hidden Markov Models (HMMs). In [4], high level descriptions of the hands' motion, shape and relative positions are extracted from video sequences, filtered via Independent Component Analysis and classified by a bank of Markov models trained for individual signs. A similar approach for Australian sign language recognition (based on geometric features and HMM classification) is presented in [5]. All these systems have shown good performances in their respective sign recognition tasks, but they cannot be directly applied to finger-spelling recognition because of the different nature of the problem. Furthermore, they depend heavily on the feature extraction task, which could degrade their performance.

Generally, in finger-spelling the discrimination between letters is based on hand and finger configurations, rather than on global hand and arm motions, as in word-level signing. Thus, global features (used in the above-mentioned systems) are not adequate for finger-spelling recognition and one needs to use more precise descriptions of the hand shape. In [6], finger-spelling recognition is addressed by hand mask extraction based on skin color and subsequent classification via nearest neighbor template matching and deterministic boosting. For the recognition of Australian finger-spelling (involving two handed global motions, unlike ASL), good results were obtained in [7], based on general geometric and motion features, recognized using HMMs. In both approaches, the feature extraction phase relies on skin color for hand region detection and performance is only guaranteed in a controlled laboratory environment, with constant-color background and similar lighting conditions during the training and testing phases.

In this paper, we introduce a method for finger-spelling recognition which is robust against cluttered background and changing lighting conditions, while being invariant to 2D similarity transformations of the signing hand. This is achieved through the unification of the two separate steps traditionally considered in recognition tasks: (i) feature extraction and (ii) classification. We propose a collaborative framework, where these two tasks are performed in parallel, so that each benefits from the knowledge and results yielded by the other one. Feature extraction is performed through variational image segmentation, which assimilates a priori knowledge regarding the most probable attributes of the signing hand, generated by the recognition process. This extra knowledge makes the segmentation robust against adverse conditions, such as cluttered backgrounds and various lighting conditions. Recognition is achieved via probabilistic inference in a multiple HMM framework, allowing the collaboration of the two processes through frame by frame integration of new segmentation results.

The remainder of this paper is organized as follows. Section 2 presents the proposed framework and details its collaborating halves: behavior inference and image segmentation. In Section 3 we describe the finger-spelling application and particularize our general framework to provide an adequate resolution. Experimental results are presented at the end of Section 3 and Section 4 concludes the paper.

2. OUR GENERAL SEGMENTATION/BEHAVIOR INFERENCE FRAMEWORK

Our general framework is based on the idea of collaboration between two processes – image segmentation and behavior [1] inference – along the target image sequence During an initial training phase, the inference process learns the dynamic probability models of typical behaviors from training data. Then, segmentation and behavior inference are run cooperatively throughout a new test image sequence. For each image, an inference step is performed, generating probabilistic prior attribute models for each behavior class. These are used by the ensuing segmentation to identify the most likely objects in the current image and subsequently provide their attributes to the next inference step. The priors offered by the inference process are based on learning from training data and are updated dynamically according to newly processed images. The

[1] By "behavior", we mean the temporal evolution of the object, as observed in the image sequence. The inference of object behavior from an image sequence requires the determination of the appropriate behavior class for each object evolution instance throughout the sequence.

most likely behavior class of each object evolution instance can be extracted at any point within the sequence from the inference process.

By the generic term "attribute" we designate a visual property of an object, definable as a functional $A(C, I)$ of the image I and of the object's segmenting contour C (A is assumed to be differentiable with respect to C). This definition includes many properties computable with boundary- and region-based functionals (e.g. position, orientation, average intensity/color, higher order statistics describing texture) and makes our framework adaptable to the needs of other behavior recognition applications.

2.1 Behavior Inference using Multiple HMMs

Given a sequence of object attribute values extracted from an image sequence, behavior inference translates to finding the best matching sequence of behavior classes. We address this task using Hidden Markov Models (HMMs) [8]. Having estimated HMM parameters from training attribute sequences, we use them to infer the behavior reflected in new image sequences, while jointly performing segmentation of these sequences, according to the intended collaboration.

An HMM [8] is a doubly embedded stochastic process, consisting of an underlying hidden process, observable via a set of stochastic processes (the HMM states) that produce a sequence of observations. In our case, the observations are the attribute values extracted from the image sequence, while the states correspond to the behavior classes. We denote the HMM states by $S = \{S_1, S_2, \ldots, S_M\}$, the state at time t by q_t and the attribute value at time t by $A(t)$. The HMM parameters are:

1. the initial state distribution $\pi = \{\pi_i\}$, with $\pi_i = P(q_1 = S_i)$, $i = 1..M$,
2. the state transition probability distribution $T = \{t_{ij}\}$, with $t_{ij} = P(q_{t+1} = S_j | q_t = S_i)$, $i, j = 1..M$, and
3. the state observation probability distributions (behavior class likelihoods):

$$P(A(t) \,|\, q_t = S_i) = P_i(A(t)),\; i = 1..M. \tag{1}$$

The class likelihoods $P_i(A(t))$ are another free parameter of our framework, adaptable to the application at hand, with the sole condition that they be differentiable with respect to $A(t)$ (to enable collaboration with the segmentation).

Many human-to-computer interaction applications require discrimination among a number of behavior types, each made up of a different succession of basic actions, belonging to different behavior classes, which are shared among the behavior types (e.g. letter classes shared among words). We model such cases via multiple HMMs, each accounting for a different behavior type and sharing the same state models, corresponding to the basic behavior classes. To perform behavior inference, we estimate the probability of an attribute sequence on the most likely state path in each HMM and choose the winner HMM (thus, behavior type) as the one with the highest probability. Its most likely state path yields the most likely succession of behavior classes for the given attribute sequence.

To estimate the most likely state path (and its probability), we run the Viterbi algorithm [8] simultaneously on all HMMs. For an observation sequence $A_{1..T}$, the Viterbi algorithm estimates – for each time step t and state S_i – the highest probability along a state sequence which accounts for the first t observations and ends in state S_i:

$$\delta_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} P(q_{1..t-1}, q_t = S_i, A_{1..t}). \tag{2}$$

To distinguish among K behavior types, we employ K HMMs, with shared states and state models $P_i(A(t))$, $i = 1..M$ and different initial $\pi^k = \{\pi_i^k\}$ and state transition probabilities $T^k = \{t_{ij}^k\}$, $k = 1..K$. We store (2) and its maximizing argument in the δ^k and ψ^k variable sets. The analysis of a sequence $A_{1..T}$ starts with variable initialization:

$$\begin{aligned} &\delta_1^k(i) = \pi_i^k \, P_i(A(1)), \\ &\psi_1^k(i) = 0, \qquad\qquad i = 1..M,\; k = 1..K. \end{aligned} \tag{3}$$

Then, for each $t = 2..T$, a recursion step is performed:

$$\begin{aligned} \delta_t^k(i) &= \max_{j=1..M} \left(\delta_{t-1}^k(j)\, t_{ji}^k\right) P_i(A(t)), \\ \psi_t^k(i) &= \arg\max_{j=1..M} \delta_{t-1}^k(j)\, t_{ji}^k,\; i = 1..M,\; k = 1..K. \end{aligned} \tag{4}$$

Finally, the probability of the attribute sequence given the most likely path in each HMM k is given by:

$$P_k^{\text{opt}} = \max_{i=1..M} \delta_T^k(i). \tag{5}$$

The winner HMM (thus, behavior type) maximizes (5)

$$k^{\text{opt}} = \arg\max_{k=1..K} P_k^{\text{opt}}. \tag{6}$$

The most likely behavior class sequence for $A_{1..T}$ can be retrieved from the δ-s and ψ-s of the winner HMM:

$$\begin{aligned} q_T^{\text{opt}} &= \arg\max_{i=1..M} \delta_T^{k^{\text{opt}}}(i), \\ q_t^{\text{opt}} &= \psi_{t+1}^{k^{\text{opt}}}(q_{t+1}^{\text{opt}}), \qquad t = T-1, T-2, \ldots 1. \end{aligned} \tag{7}$$

Our innovation is to couple behavior inference and segmentation by using the probability estimates of the Viterbi algorithm at each step to guide the segmentation of the corresponding image. To this end, we run the algorithm and segmentation in an interleaved manner along the image sequence, using as observations the attributes of newly segmented images as soon as they become available. Suppose we have completed step $t-1$ of our framework, so that $A_{1..t-1}$ and $\delta_{t-1}^k(j)$, $j = 1..M$, $k = 1..K$ are available. We invest into the segmentation of $I(t)$ the maximum amount of a priori knowledge given by the inference process:

1. the predictions of each class i for the next attribute $A(t)$; i.e., the likelihood functions $P_i(A(t))$ (1), and
2. our relative confidence in the class predictions, given by the maximum probability of reaching state S_i at time step t, after having observed attributes $A_{1..t-1}$. This probability can be estimated as:

$$\begin{aligned} w_t(i) &= \max_{k=1..K} \max_{q_{1..t-1}} P(q_{1..t-1}, q_t = S_i, A_{1..t-1} \,|\, k) \\ &= \max_{\substack{j=1..M \\ k=1..K}} \delta_{t-1}^k(j) t_{ji}^k. \end{aligned} \tag{8}$$

We define the prior information offered by class i about the next attribute $A(t)$ as the product of the two quantities above:

$$\begin{aligned} \delta_t(A(t), i) &= w_t(i)\, P_i(A(t)), \qquad\qquad i = 1..M \\ &= \max_{k=1..K} \delta_t^k(A(t), i). \end{aligned} \tag{9}$$

2.2 Variational Image Segmentation

Motivated by successful segmentation approaches using prior information [9, 10], we formulate segmentation in a variational framework which incorporates the probabilistic behavior class priors $\delta_t(A(t), i)$ via a competition approach. In this way, the segmented object belongs to the class which best accounts for its generation, given the image evidence. Having run our joint segmentation / behavior inference framework on the first $t-1$ frames of an image sequence, we segment $I(t)$ by minimizing the following energy functional:

$$E(C, \mathcal{L}, I(t)) = E_{\text{data}}(C, I(t)) + \alpha E_{\text{prior}}(C, \mathcal{L}, I(t)), \tag{10}$$

where C is the segmenting contour, $\mathcal{L} = (L_1, \ldots L_M)$ is the set of labels (defined below) and α is a positive weighing constant. Energy $E_{\text{data}}(C, I(t))$ encapsulates image-related constraints on the contour C, and can be any boundary- or region-based segmentation energy suitable for the application at hand (e.g. [11]). Energy $E_{\text{prior}}(C, \mathcal{L}, I(t))$ is:

$$E_{\text{prior}}(C, \mathcal{L}, I(t)) = -\sum_{i=1}^{M} \log\left(\delta_t(A(C, I(t)), i)\right) L_i^2 + \beta \left(1 - \sum_{i=1}^{M} L_i^2\right)^2 . \quad (11)$$

with β – a positive constant. It contains the negative logarithms of the prior probabilities (9), which through energy minimization will lead to the maximization of the respective probabilities. Each prior carries a label factor L_i^2, which controls its contribution to segmentation according to its relative probability with respect to the other priors. The label L_i is a scalar variable that varies continuously between 0 and 1 during energy minimization and converges either to 1 (for the winning prior, whose probability has thus been maximized through segmentation) or to 0 (for the other priors, which have thus been annulled). Competition among priors is enforced by the soft constraint that the label factors should sum to 1, introduced by the last term in (11).

We minimize (10) simultaneously with respect to the segmenting contour C and the labels $\mathcal{L}$ using the calculus of variations and gradient descent. The corresponding equations are not included here due to space limitations.

2.3 Summary

To sum up, our general framework for segmentation and behavior inference consists of the following:

- **Training phase:** estimate parameters of the HMMs from training attribute sequences, according to [8].
- **Testing phase:** perform joint segmentation and behavior inference on new attribute sequences $A_{1..T}$:
  1. Segment first image in the sequence $I(1)$ (manually or using only $E_{\text{data}}(C, I(1))$ in (10).
  2. Extract attribute $A(1) = A(C, I(1))$.
  3. Initialize δ and ψ according to (3).
  4. For t = 2..T
     - Compute $w_t(i)$, $i = 1..M$ according to (8).
     - Segment image $I(t)$ using energy (10), with $\delta_t(A(C, I(t)), i)$ given by (9).
     - Extract attribute $A(t) = A(C, I(t))$.
     - Compute $\delta_t^k(i)$ and $\psi_t^k(i)$, $i = 1..M$, $k = 1..K$ using (4).
  5. Estimate winner HMM and infer behavior type using (6).
  6. Backtrack to infer the behavior class of each attribute instance in $A_{1..T}$ using (7).

3. FINGER-SPELLING RECOGNITION

In the following, we perform finger-spelling recognition using our collaborative segmentation/behavior inference framework. We first describe our application. Then, we particularize our framework using likelihood and segmentation models adapted to our application. Finally, we present the obtained results.

3.1 Application description

For our application, we use the manual alphabet of the French-speaking part of Switzerland (Suisse Romande) (see [12]). Our goal is to perform finger-spelling recognition on a 30-word vocabulary, containing country names (Table 1).

With the support of the Swiss Federation for the Hearing-Impaired [12], we have acquired a data base containing image sequences of a hearing-impaired person finger-spelling the above mentioned words. Acquisition has been performed both in ideal conditions (contrasting background, low speed gesturing), for training purposes, and realistic ones (cluttered background, normal speed gesturing), for testing purposes.

Albania	Algeria	Armenia	Austria	Belarus
Belgium	Burundi	Croatia	Denmark	Ecuador
Eritrea	Estonia	Finland	Georgia	Germany
Hungary	Iceland	Lebanon	Lesotho	Liberia
Moldova	Namibia	Nigeria	Romania	Senegal
Somalia	Tunisia	Ukraine	Uruguay	Vietnam

Table 1: Vocabulary of our finger-spelling application

3.2 Solution using the proposed framework

For this application, we use the hand contour as attribute $A(C, I) = C$, represented by a level set function (LSF) $\phi : \Omega \to \mathbb{R}$, where Ω is the image domain [13].

The words in our vocabulary constitute our behavior types and are each modeled by an individual HMM. Letters are the common basic components of all words and are modeled as shared states (behavior classes) of our HMMs.

The likelihood model $P_i(\phi)$ for each class i adapts dynamically to new image content and relies on a shape distance function, motivated by [14], between the segmenting contour and a prior contour corresponding to that class. The prior contours are computed via principal components analysis (PCA) from specific training data for each class. They evolve during segmentation so as best to match image information, within class constraints imposed by the PCA.

Based on the training LSFs for a class, we approximate a new LSF $\hat{\boldsymbol{\phi}}$ from that class via PCA as:

$$\hat{\boldsymbol{\phi}} = \overline{\boldsymbol{\phi}} + \mathrm{E\,c}, \quad (12)$$

where $\overline{\boldsymbol{\phi}}$ is the mean of the training LSFs, E is a matrix whose columns are a reduced set of p PCA eigenvectors and c is the p-dimensional vector of eigencoefficients.

Our shape distance function between the current segmenting contour ϕ and the prior contour $\hat{\phi}$ is given by:

$$d(\phi, \mathrm{c}, \boldsymbol{\tau}) = \iint_{\Omega} \left(\hat{\phi}^2 |\nabla\phi| \delta(\phi) + \phi^2 |\nabla\hat{\phi}| \delta(\hat{\phi})\right) dx\, dy. \quad (13)$$

Here, δ is the Dirac function and $\hat{\phi}$ is the continuously interpolated LSF of the prior contour, obtained from (12):

$$\hat{\phi}(\mathrm{c}, \boldsymbol{\tau}) \mid_{(x,y)} = \frac{1}{s} \left(\overline{\boldsymbol{\phi}}(h_{\boldsymbol{\tau}}(x,y)) + \mathrm{E}(h_{\boldsymbol{\tau}}(x,y))\, \mathrm{c}\right). \quad (14)$$

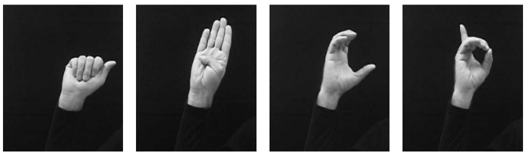
Here $\boldsymbol{\tau} = \{s, \theta, T_x, T_y\}$ are the parameters of a similarity transformation which aligns the prior with contour ϕ:

$$h_{\boldsymbol{\tau}}\left([x \;\; y]^T\right) = s \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix}. \quad (15)$$

Since $\iint_{\Omega} |\nabla\phi| \delta(\phi)\, dx\, dy$ represents the length of the zero level set of ϕ and the LSFs are represented as signed distance functions, we readily observe that the first term of (13) approximates the minimal Euclidian distance to the prior contour, integrated along the segmenting contour. The second term of (13) exchanges the roles of ϕ and $\hat{\phi}$ relative to the first term, making the distance function symmetric and thus suitable for use in classification. Based on this distance function, we define the likelihood of the segmenting contour represented by ϕ, for time t (image $I(t)$) and class i as

$$P_i(\phi(t)) = e^{-d(\phi(t), \mathrm{c}^i(t), \boldsymbol{\tau}^i(t))}. \quad (16)$$

We use the piecewise constant Mumford-Shah model [11] to guide the evolution of the main contour ϕ and prior contours



Letter A Letter B Letter C Letter D

Figure 1: Sample images (and corresponding behavior classes) from training sequences used in our application.

$\hat{\phi}_i(\mathrm{c}^i, \boldsymbol{\tau}^i)$, in terms of their parameters c^i and $\boldsymbol{\tau}^i$:

$$\begin{aligned} E_{\text{data}}(\phi,\, \mathrm{c}^{i=1..M}, \boldsymbol{\tau}^{i=1..M}) &= \nu \iint_\Omega |\nabla H(\phi)|\, dx\, dy \\ + \quad & \iint_\Omega (I-\mu_{\phi+})^2 H(\phi) + (I-\mu_{\phi-})^2 H(-\phi)\, dx\, dy \\ + \sum_{i=1}^{M} & \iint_\Omega (I-\mu_{\hat{\phi}_i+})^2 H(\hat{\phi}_i) + (I-\mu_{\hat{\phi}_i-})^2 H(-\hat{\phi}_i)\, dx\, dy. \end{aligned} \tag{17}$$

Here H is the Heaviside function and $\mu_{\phi+}$, $\mu_{\hat{\phi}_i+}$ and $\mu_{\phi-}$, $\mu_{\hat{\phi}_i-}$ are the mean image intensities over the positive, respectively negative, regions of the LSFs ϕ and $\hat{\phi}_i$. The first term of (17) imposes smoothness of the contour ϕ.

Using likelihoods (16) for the priors $\delta_t(\phi, i)$ in (9), the prior energy becomes:

$$\begin{aligned} E_{\text{prior}}(\phi, \mathcal{L},\, \mathrm{c}^{i=1..M}, \boldsymbol{\tau}^{i=1..M}) &= \sum_{i=1}^{M} \Bigg(-\log w_t(i) \\ + d(\phi(t), \mathrm{c}^i(t), \boldsymbol{\tau}^i(t)) \Bigg) L_i^2 &+ \beta \left(1 - \sum_{i=1}^{M} L_i^2\right)^2 . \end{aligned} \tag{18}$$

In practice, to decrease the computational burden during segmentation, we used only the top 3 most probable letter priors (estimated with (8)) to guide segmentation, instead of the 20 available letter priors. This pruning strategy did not affect recognition performance, while diminishing segmentation time and improving convergence towards the optimal prior.

The total energy (10), summing (17) and (18), is minimized via the calculus of variations and gradient descent, yielding evolution equations for the LSF ϕ, the labels $\mathcal{L}$, the PCA and alignment parameters c^i and $\boldsymbol{\tau}^i$, $i = 1..M$.

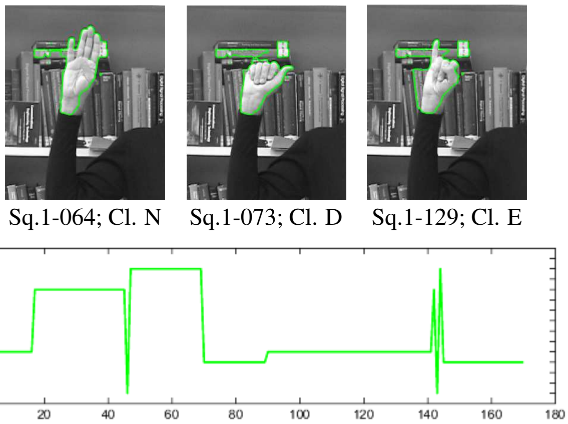
3.3 Training the model

We have trained our model using image sequences of each vocabulary word from the acquired database, where the gesturing person was filmed on a dark, contrasting background and the gestures were performed at slow speed. Figure 1 presents images from the training sequences.

First, the gesturing hand has been segmented in each training sequence and the resulted contours have been assigned to their respective letter classes and aligned with respect to similarity transformations using genetic algorithms [15]. Subsequently, a separate HMM was trained for each vocabulary word [8]. The observation probabilities for the shared HMM states have been learned by PCA ($p = 20$) from the contours of each letter class.

3.4 Experimental Results

We tested the resulting implementation of our framework on image sequences of the same person finger-spelling words from the vocabulary, this time in realistic conditions, involving a cluttered background, normal gesturing speed and changed lighting conditions with respect to the training image sequences. Despite the complexity of the task, the results are accurate in terms of the recognized words, due to the infusion of knowledge about the dynamics of vocabulary words via our collaborative framework. Figure 3 presents examples of cooperative segmentation and behavior inference on two image sequences, which have been correctly classified as the words "Albania" and "Belarus", respectively. The dynamical PCA-based prior models have adapted to significant shape variations within behavior classes, allowing the segmentation of the hand in a difficult case of cluttered background. The frame-wise behavior inference results for these sequences, yielded by the winner HMMs, are presented on the bottom of in Fig. 3 and correspond to our understanding of the sequences in terms of the executed gestures. In contrast, using the traditional (sequential) approach for recognition, i.e. variational segmentation without prior models, followed by inference, produces erroneous results. For instance, Fig. 2 shows how the "Albania" sequence was miss-classified as "Iceland", because the segmentation has been side-tracked by the cluttered background.



Sq.1-064; Cl. N Sq.1-073; Cl. D Sq.1-129; Cl. E

Frame number

Figure 2: Erroneous results using *sequential* segmentation and behavior inference on the "Albania" sequence, miss-classified as "Iceland".

4. CONCLUSION

We have introduced a novel approach for finger-spelling recognition, based on a collaborative framework that fuses feature extraction and classification. The advantage of our framework is the sharing of information between the two processes, which is mutually beneficial. Feature extraction is based on variational image segmentation, which allows the introduction of prior models of the hand, resulted from classification. Classification is performed through behavior inference using HMMs, enabling the step-by-step incorporation of new attributes extracted from the image sequence. Our approach yields good results both in terms of image segmentation and of gesture recognition, proving robustness in difficult situations, involving cluttered background and changing lighting conditions.

REFERENCES

[1] C. Wang, W. Gao, and J. Ma, "A real-time large vocabulary recognition system for Chinese Sign Language," in *Gesture Workshop*, pp. 86–95, 2001.

[2] C. Vogler and D. Metaxas, "Handshapes and movements: Multiple-channel ASL recognition," in *Gesture Workshop'03, in Lecture Notes in Artificial Intelligence*, pp. 247–258, 2004.

[3] 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(12), pp. 1371–1375, 1998.

[4] R. Bowden, D. Windridge, T. Kadir, A. Zisserman, and M. Brady, "A linguistic feature vector for the visual interpre-

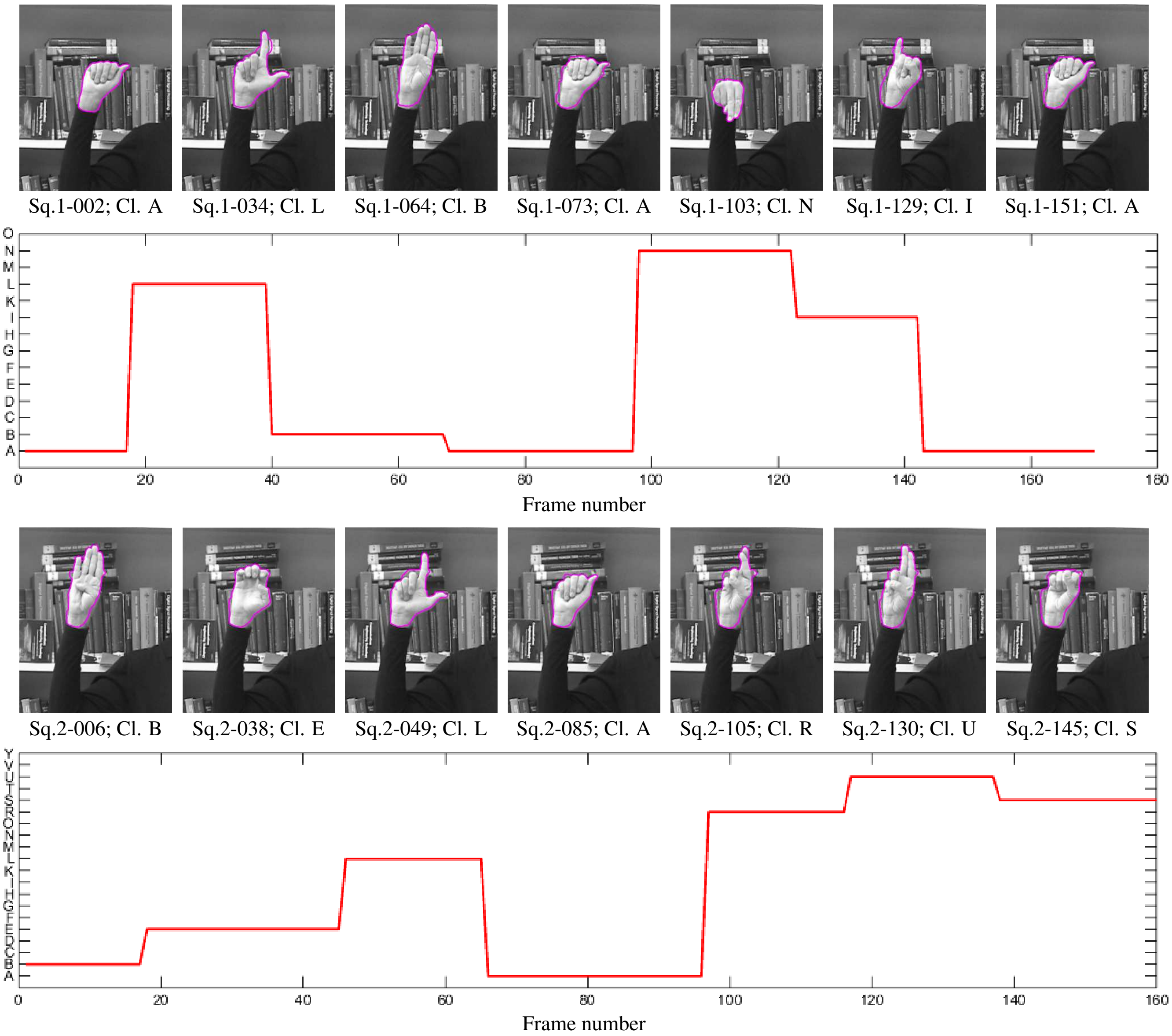


Figure 3: Correct segmentation and behavior inference using our framework, demonstrated on two test sequences representing the words "Albania"(rows 1-2) and "Belarus" (rows 3-4).

tation of sign language," in *The 8th European Conference on Computer Vision*, pp. 391–401, 2004.

[5] E. J. Holden, G. Lee, and R. Owens, "Automatic recognition of colloquial Australian sign language," in *IEEE Workshop on Motion and Video Computing*, 2005.

[6] R. Lockton and A. Fitzgibbon, "Real-time gesture recognition using deterministic boosting," in *British Machine Vision Conference*, 2002.

[7] P. Goh and E. J. Holden, "Dynamic fingerspelling recognition using geometric and motion features," in *IEEE Int. Conf. on Image Processing*, pp. 2741–2744, 2006.

[8] L. R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77(2), 1989.

[9] N. Paragios and M. Rousson, "Shape priors for level set representations," in *European Conference in Computer Vision*, vol. 2, pp. 78–92, 2002.

[10] D. Cremers, N. Sochen, and C. Schnör, "Multiphase dynamic labeling for variational recognition-driven image segmentation," in *European Conf. on Computer Vision*, vol. 3024, pp. 74–86, 2004.

[11] T. Chan and L. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10(2), pp. 266–277, 2001.

[12] FSS, "Fédération Suisse des Sourds," 2007. http://www.sgb-fss.ch/.

[13] S. Osher and J. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on the Hamilton-Jacobi formulation," *Journal of Computational Physics*, vol. 79, pp. 12–49, 1988.

[14] X. Bresson, P. Vandergheynst, and J.-P. Thiran, "A variational model for object segmentation using boundary information and shape prior driven by the Mumford-Shah functional," *International Journal of Computer Vision*, vol. 28, pp. 145 – 162, July 2006.

[15] L. Davis, *Handbook of Genetic Algorithms*. New York, NY, USA: Van Nostrand Reinhold, 1991.