Leveraging Intra-Class Variations to Improve Large Vocabulary Gesture Recognition

Christopher Conly, Alex Dillhoff, Vassilis Athitsos
Department of Computer Science and Engineering
University of Texas at Arlington
christopher.conly@mavs.uta.edu, alex.dillhoff@mavs.uta.edu, athitsos@uta.edu

***Abstract*—Large vocabulary gesture recognition using a training set of limited size is a challenging problem in computer vision. With few examples per gesture class, researchers often employ exemplar-based methods such as Dynamic Time Warping (DTW). This paper makes two contributions in the area of exemplar-based gesture recognition: 1) it introduces Multiple-Pass DTW (MP-DTW), a method in which scores from multiple DTW passes focusing on different gesture properties are combined, and 2) it introduces a new set of features modeling intra-class variation of several gesture properties that can be used in conjunction with MP-DTW or DTW. We demonstrate that these techniques provide substantial improvement over DTW in both user-dependent and user-independent experiments on American Sign Language (ASL) datasets, even when using noisy data generated by RGB-D skeleton detectors. We further show that using these techniques in a large vocabulary system with a limited training set provides significantly better results compared to Long Short-Term Memory (LSTM) network and Hidden Markov Model (HMM) approaches.**

I. Introduction

Gesture and sign language recognition is a challenging research field in computer vision. Popular probabilistic methods like Hidden Markov Models (HMM) [1] and Conditional Random Fields (CRF) [2] require large training sets to learn probability distributions. This requirement often limits the vocabulary size of such systems. When a large vocabulary is desired, however, time and fiscal constraints may force researchers to restrict the size of the training set and thus limit the techniques that can be used for classification. As using few examples per gesture class precludes the use of many statistical and machine learning methods, researchers are often limited to exemplar-based recognition and similarity measures.

In such cases, Dynamic Time Warping (DTW) [3] is often used on hand position or other information to generate scores that serve as a measure of similarity to training examples [4], [5]. DTW is improved with the use of a well-designed feature vector that includes more than hand positions to represent the state of a gesture at each point in time [6]. The performance of DTW-based recognition, however, can suffer due to variations in gesture performance inherent in user-independent systems. The two methods presented in this paper address this problem.

Both methods are based on natural variations that occur when multiple persons perform the same gesture. The intuitive notion behind the methods is that individuals will perform the same gesture in different positions and orientations in the gesture space or at different sizes, and the goal becomes to use knowledge of these inherent variations to improve gesture recognition accuracy.

The first method models this intra-class variation (ICV) in the geometric and positional properties of same-class gestures to provide indications of likelihood that a test gesture belongs to the same class as a training gesture, given their differences in these properties. The likelihoods become features that, when linearly combined and used in conjunction with DTW scores, provide a better indication of gesture similarity.

The second method, Multiple-Pass DTW (MP-DTW), involves generating multiple DTW scores for a test gesture. The DTW technique on which this work is based, detailed in section III-A, expresses hand positions in a single coordinate system centered on the head and normalizes the gesture size based on the face size. This works well when the testing and training subjects perform the gesture in roughly the same position and at the same size. In practice, there is wide variation in both where a gesture is performed and how large the space it occupies. It makes sense to adjust both this coordinate system and how the gesture is sized, and to combine weighted scores from multiple DTW passes using the new centering and resizing techniques.

We show that either of these contributions alone or in combination can provide a substantial improvement in accuracy over DTW, even when using noisy and unstable RGB-D skeletal data. We then compare these methods to HMM and LSTM network approaches [7]–[9] to demonstrate the benefit of using them in gesture recognition systems that comprise a large vocabulary but have a small training set size.

II. Related Work

Most recent work has been in action and activity recognition, some from static images [10], others from video [11]. These works tend to focus on classifying small vocabularies of general actions, for example recognizing golf vs. gymnastics, rather than one gymnastics move vs. another. This is somewhat analogous to identifying a video as sign language vs. juggling, rather than identifying specific signs. Some action recognition works do test their methods on gesture datasets [12], [13], but the vocabularies are limited, and the methods are generally not directly applicable to our large vocabulary gesture sets.

Other research focuses on general gesture recognition. The gesture sets may be created specifically for this task and can be chosen so as to minimize similarity between classes. LSTM networks, one of our comparison methods, have seen success, and Google has incorporated one into Android to recognize keyboard gestures [14]. With the recent ChaLearn Gesture

Challenge [15], there have been a number of works in one-shot learning, in which a single training example is used per class. Wan et al. propose utilizing the multimodal data of the Kinect to create a 3D Enhanced Motion Scale-Invariant Feature Transform (3D EMoSIFT) to describe both motion and appearance for use in a Bag of Features style approach [16]. Konečný et al. propose using Dynamic Time Warping and a combination of Histogram of Oriented Gradients (HOG) features to describe appearance of the depth images and Histogram of Optical Flow (HOF) features to describe the motion [17]. Jiang et al. propose a 3 classifier hierarchical approach that progressively eliminates candidate matches with each layer [18]. The experiments for the ChaLearn gesture challenge, however, were user dependent. The multiple performances of the gestures occur in the same position in the video frame, and many of the methods take advantage of this fact and use global level features on entire video frames. These assumptions are unrealistic in our work, and it is unclear how well these methods generalize to handle inter-user variation and scale from the 8 to 12 gesture ChaLearn vocabularies to ours of 1,113.

A third focus is on developing methods for well-established gesture sets, such as sign languages. One branch of work is in continuous sign language recognition and fingerspelling, or the spelling out of words with a signed alphabet. Kim et al. propose a method to break a video into variable length segments, using letter transition probabilities, hand shape similarities scores, and a semi-Markov CRF to identify the string of letters in fingerspelling videos [19]. Other research focuses instead on classification of individually segmented signs. One popular intuitive method is to segment a sign into motion or other types of sub-units and then use an HMM to model the temporal changes in sub-units throughout each sign. Cooper et al. provide a comparison of two sub-unit methods, experimenting with both an HMM and Sequential Pattern Boosting [20]. Using an HMM on frames their method designates as high-ranked key frames, Wang et al. achieve good results in user-independent tests on their large Kinect dataset of Chinese Sign Language (DEVISIGN) using a vocabulary of 1,000 signs [21]. HMMs work well with enough training examples to learn the transition probabilities, but our experiments show that 3 examples per class, as found in our dataset, is insufficient.

Some works also approach the idea of class variability modeling. Reyes et al. use inter- and intra-class variability in joint positions to weight the contribution each joint has in DTW scoring for the classification of 5 action categories [22]. To handle variations inherent in performances of same-class gestures by different subjects, Yao et al. generate a group of likelihood maximization-based classifiers and use the best one for each subject based on personalization data [23]. Bautista et al. use intra-class variability in gesture feature vectors to learn a Gaussian Mixture Model and extend DTW to be probabilistic [24]. Our method instead models variations in geometric properties of whole gesture trajectories to improve results. It may be beneficial to employ multiple approaches.

III. METHOD

The presented methods build on past work in ASL recognition. The base method used as a measure of gesture similarity is DTW, a dynamic programming technique that creates an optimal alignment of two sequences [3], in this case, the hand trajectories of test and model signs. It has the benefit of being able to warp the temporal dimension of the series, so that is can properly align gestures performed at different speeds. The score provided by DTW is a measurement of error in the alignment, so that a lower score indicates a better match.

Rather than solely using the hand positions to describe the trajectories, we modify the feature vector introduced in [6] to use information available from RGB-D output and then use multiple passes through DTW to generate several scores per example sign. Each pass focuses on a different gesture property and size normalization technique. We also introduce a new set of features that describe the likelihood of the measured variations between the test and example sign in several geometric and positional properties. The scores from the features and from multiple DTW passes can then be linearly combined to improve recognition accuracy.

A. Base DTW Method

To represent a sign, a feature vector based on 2D hand position information is built for each video frame that describes what is occurring at that point in time. For the base DTW pass, the hand positions are first expressed in a face-centric coordinate system. For one-handed signs, the position of the non-dominant hand is set to $(0, 0)$ so as not to contribute to the DTW score. The following features compose the vectors for each frame t of sign video X:

1) $L_d(X, t)$ and $L_{nd}(X, t)$: Pixel position of the dominant and non-dominant hands.
2) $L_\delta(X, t) = L_d(X, t) - L_{nd}(X, t)$: Position of the dominant hand relative to the non-dominant hand.
3) $O_d(X, t)$ and $O_{nd}(X, t)$: Motion direction, expressed as unit a vector, from frame $t-1$ to frame $t+1$ for the dominant and non-dominant hands.
4) $O_\delta(X, t)$: Direction of change for L_δ from frame $t-1$ to frame $t+1$, expressed as a unit vector.

The feature vectors for each frame are combined into a single matrix to describe the sign. For experiments using manual annotations of the hand positions, the sign is size-normalized so that the diagonal of the face bounding box is 1. For experiments using RGB-D skeleton detector-provided hand positions, the sign is size-normalized so that the average shoulder width during the sign is 1.25. Finally, using bicubic interpolation on the feature matrix, the frame length is normalized to 25 frames.

DTW creates a minimal cost warping path between two signs by effectively matching frame by frame what is occurring in the test and model videos, as described by the feature vectors. The score for matching model sign M to query sign Q given any warping path $W = ((q_1, x_1), ..., (q_{|W|}, x_{|W|}))$ of length $|W|$ is the summation of the individual costs to match query frames q_i to model frames m_i:

$$C(W, Q, M) = \sum_{i=1}^{|W|} c(Q_{q_i}, M_{m_i}),$$

where the local cost of matching Query frame number q_i to model frame number m_i is a weighted linear combination of

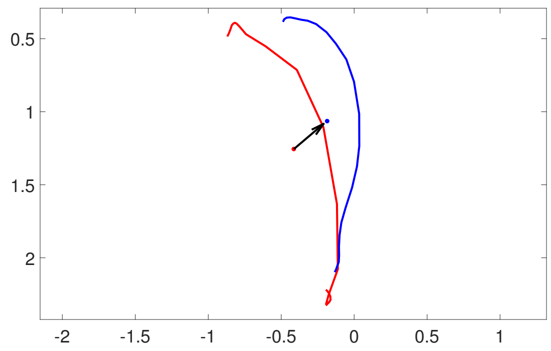


Fig. 1. 2D property example. Shown are trajectories for two examples of the same sign. The measured variation is shown by the black arrow drawn from the centroid of one trajectory to the other.

squared Euclidean distances between features:

$$\begin{aligned} c(Q_{q_i}, M_{m_i}) = & s_1 \| L_d(Q, q_i) - L_d(M, m_i) \|^2 & + \\ & s_2 \| L_{nd}(Q, q_i) - L_{nd}(M, m_i) \|^2 & + \\ & s_3 \| L_\delta(Q, q_i) - L_\delta(M, m_i) \|^2 & + \\ & s_4 \| O_d(Q, q_i) - O_d(M, m_i) \|^2 & + \\ & s_5 \| O_{nd}(Q, q_i) - O_{nd}(M, m_i) \|^2 & + \\ & s_6 \| O_\delta(Q, q_i) - O_\delta(M, m_i) \|^2 . & \end{aligned}$$

Weights $\{s_1, ..., s_6\}$ are empirically determined on a validation set. The base DTW score D_b between query Q and example M is provided by the lowest cost of all warping paths:

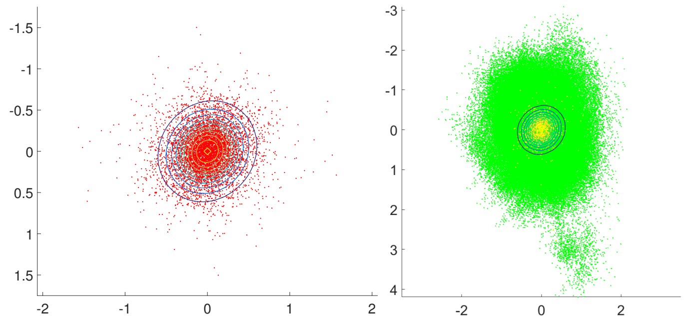
$$D_b(Q, M) = \min_W C(W, Q, M).$$

A separate score is calculated for each of the three examples per sign class, and the lowest score for each class is used for sign ranking purposes. To these baseline scores can be added scores from the intra-class variation modeling and MP-DTW methods described in sections III-B and III-C. Hand shape is not considered and is left for future work.

B. Intra-Class Variation Modeling

This section introduces the intra-class variation modeling (ICVM) of several hand trajectory geometric properties and describes the new features that are generated from the models. These features give an indication of likelihood that a test sign would vary in these geometric aspects from a given model sign by the observed amount. Two sets—*LB1113* and *TB1113*—of one example each of 1,113 unique signs obtained from the ASLLVD [6] and a third set *GB1113* of the same signs, obtained from an alternate source, are used to train the variation models. *LB1113* and *TB1113* are each performed by a single signer, while *GB1113* consists of signs performed by multiple signers. Once the signs are expressed in the size-normalized, face-centric coordinate system described in section III-A, we measure the difference in the properties between each sign of the same class. For example, figure 1 shows the trajectory for two signs of the same class. The measured variation is between the centroids of the two trajectories. The black arrow represents this difference vector.

Once the differences between all the same-class examples have been collected for the measured properties, a separate Gaussian is learned for each property to model the variation. Figure 2a plots the difference vectors as points for the convex hull centroid property of the dominant hand trajectory. It is clear that the single Gaussian that is overlaid is sufficient to model the variation in the property.



(a) 2D Property Gaussian (b) Variation Plot

Fig. 2. Left: Intra-class variation plot for the centroid of the convex hull encompassing the dominant hand trajectory. The learned Gaussian model is overlaid. Right: A plot as 2D points of the measured differences of all test signs from each example sign for the dominant hand trajectory convex hull centroid property. The differences from same-class signs are plotted in yellow and from dissimilar classes in green. The Gaussian model learned during training is overlaid to show it generalizes well to test sets.

The variations in the following sign properties are modeled, separately for the dominant (d) and non-dominant (nd) hands in two-handed signs. The trajectory bounding box is defined as the box that extends from the leftmost to the rightmost hand position and from the topmost to the bottommost hand position throughout the sequence of video frames. Some properties, as indicated, are derived from the convex hull encompassing the set of hand positions throughout the sign.

1) γ_d and γ_{nd}: center of hand trajectory bounding box.
2) ψ_d and ψ_{nd}: width of trajectory bounding box.
3) η_d and η_{nd}: height of trajectory bounding box.
4) α_d and α_{nd} : position of the hand in the first frame.
5) ω_d and ω_{nd}: position of the hand in the last frame.
6) λ_d and λ_{nd}: eigenvalue corresponding to the eigenvector describing the principle orientation of the trajectory.
7) σ_d and σ_{nd}: smallest eigenvalue subtracted from the largest eigenvalue: an indication of the strength in the trajectory orientation.
8) π_d and π_{nd}: perimeter of the trajectory convex hull.
9) ρ_d and ρ_{nd}: area of the trajectory convex hull.
10) ξ_d and ξ_{nd}: centroid of the trajectory convex hull.

Figure 2b plots the difference vectors of a set of test signs from the model signs as 2D points. The variation from same-class signs is plotted in yellow, while the variation from different-class signs is plotted in green. The Gaussian learned on the training set is overlaid on the figure, and it can be seen that it generalizes well to the test set.

To generate a feature value $\phi_i(Q, M)$ for a given query sign Q using property i, the 1D difference x or 2D difference (x, y) is measured between Q and M for property i. The Gaussian for property i is evaluated using learned parameters (μ_i, σ_i) to calculate the feature value for 1D and 2D properties,

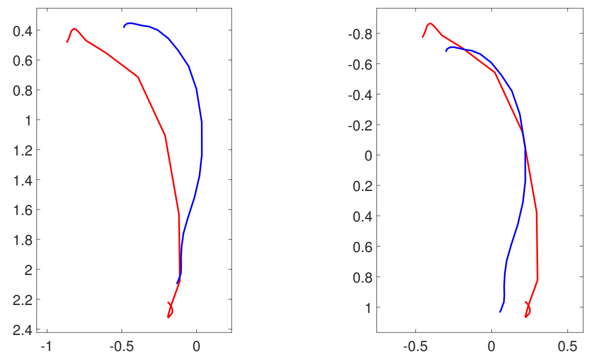


Fig. 3. Motivation for MP-DTW. Left: gestures aligned on the face. Right: gestures aligned on the convex hull centroid. Using the centroid potentially gives a better DTW gesture alignment than the face. Combining multiple alignment methods results in better recognition accuracy.

respectively:

$$\phi_i(Q, M) = \frac{1}{\sigma_i\sqrt{2\pi}} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}}$$

$$\phi_i(Q, M) = \frac{1}{2\pi\sigma_{x_i}\sigma_{y_i}} e^{-\frac{(x-\mu_{x_i})^2}{2\sigma_{x_i}^2}+\frac{(y-\mu_{y_i})^2}{2\sigma_{y_i}^2}}$$

The feature values for multiple properties can be weighted and added to the base DTW scores to improve accuracy.

The ICVM features and corresponding weights are trained using the *GB1113* dataset as the query set and the *LB1113* and *TB1113* datasets as the model sets. A separate set of features and weights are learned for one-handed and two-handed signs as follows. Given the set of available feature properties

$$F = \{\gamma_d, \gamma_{nd}, \psi_d, \psi_{nd}, \eta_d, \eta_{nd}, \alpha_d, \alpha_{nd}, \omega_d, \omega_{nd}, \lambda_d, \lambda_{nd}, \sigma_d, \sigma_{nd}, \pi_d, \pi_{nd}, \rho_d, \rho_{nd}, \xi_d, \xi_{nd}\},$$

a subset of properties $N \subseteq F$ is simultaneously selected and weighted in a greedy manner: while an accuracy improvement above a threshold τ is achieved or there are remaining properties to be chosen, the property and weight combination (ν, β) that provides the best improvement when combined with previously selected features and base DTW scores is then included in N and removed from F. For one-handed signs, only the properties for the dominant hand are considered.

C. Multiple-Pass Dynamic Time Warping

Since different signers will likely vary the position and size of a gesture, it is beneficial to use multiple alignment and size normalization methods for recognition. Figure 3 illustrates this benefit. The trajectories of two examples of the same sign class are shown. On the left, the trajectories are expressed in the face-centric coordinate system of the base method. On the right, the convex hull centroids become the origin of the coordinate system and a better alignment is achieved.

For the following two sets of gesture properties, d indicates the property applies to the dominant hand trajectory, nd to the non-dominant hand trajectory, and c to the combined dominant and non-dominant hand trajectories. The set I of properties available for centering the gesture includes:

1) η: coordinates of the head.
2) Γ_d, Γ_{nd}, Γ_c: center of trajectory bounding box.
3) A_d and A_{nd} : position of the hand in the first frame.
4) Ω_d and Ω_{nd}: position of the hand in the last frame.
5) Ξ_d, Ξ_{nd}, Ξ_c: centroid of the trajectory convex hull.
6) M_d, M_{nd}, M_c: mean hand position during gesture.

The set K of properties available to use for resizing the gestures includes:

1) Θ: face diagonal size
2) Ψ_d, Ψ_{nd}, Ψ_c: width of trajectory bounding box.
3) H_d, H_{nd}, H_c: height of trajectory bounding box.
4) Δ_d, Δ_{nd}, Δ_c: diagonal of trajectory bounding box.
5) Λ_d, Λ_{nd}, Λ_c: diameter of the trajectory, defined as as the largest distance between any two hand positions throughout the gesture.

To generate a feature value $\zeta_j(Q, M)$ for query Q and model M using centering and size-normalization property pair j, Q and M are centered and resized using j, and DTW is run to obtain a score. The score becomes the feature value that can be weighted and combined with other DTW passes.

$$\zeta_j(Q, M) = D_j(Q, M)$$

Using the *GB1113* dataset as the query set and *LB1113* and *TB1113* as model sets, a separate set of centering and size normalization properties are trained for one-handed and two-handed signs as follows. Given the set of centering properties

$$I = \{\eta, \Gamma_d, \Gamma_{nd}, \Gamma_c, A_d, A_{nd}, \Omega_d, \Omega_{nd}, \Xi_d, \Xi_{nd}, \Xi_c, M_d, M_{nd}, M_c\}$$

and the set of size normalization properties

$$K = \{\Theta, \Psi_d, \Psi_{nd}, \Psi_c, H_d, H_{nd}, H_c, \Delta_d, \Delta_{nd}, \Delta_c, \Lambda_d, \Lambda_{nd}, \Lambda_c\}$$

a subset of centering and resizing properties $\Upsilon = (\upsilon_1, ..., \upsilon_{|\Upsilon|})$, where each $\upsilon = (\iota_m, \kappa_n) \in I \times K$, is simultaneously selected (with replacement) and weighted in a greedy manner: while accuracy improvement is above a threshold τ, each combination of centering property ι_m and size normalization property κ_n is used to center and resize the sign for the DTW pass to obtain a score. The property combination and score weight (υ, β) that provide the best accuracy improvement when linearly combined with the base DTW score, previously selected MP-DTW feature scores, and ICVM features is included in Υ. For one-handed signs, only features for the dominant hand are considered.

D. Combining MP-DTW Scores and ICVM Features

To generate a final score for an alignment between query sign Q and model sign M, the scores from DTW, MP-DTW, and ICVM features are linearly combined. Given base DTW score D_b, the set N of ICVM features, and the set Z of MP-DTW scores:

$$S(Q, M) = D_b(Q, M) + \sum_{i=1}^{|N|} \beta_i \phi_i(Q, M) + \sum_{j=1}^{|Z|} \beta_j \zeta_j(Q, M),$$

where $\phi_i \in N$ and $\zeta_j \in Z$. Though the lowest final score of the three examples for each sign class is used for ranking purposes in our experiments, it is possible to combine them into a single score. Since this version of DTW provides an error measurement of the alignment of two gestures, a lower score indicates a better match.

IV. Experiments and Results

In this section, we demonstrate the significant improvement in accuracy that ICVM features and MP-DTW provide using both manual annotations and the noisy joint position data generated by Kinect skeleton detectors. Our results show that systems using a large vocabulary with few training examples per gesture class benefit from incorporating one or both of the techniques. It is clear that our method outperforms popular methods that rely on large training sets or small vocabularies. To evaluate ICVM and MP-DTW, we performed a series of user-dependent and user-independent experiments on two datasets using manually provided hand positions and Kinect joint positions.

We used the *GB1113*, *LB1113*, *TB1113* datasets as models for the experiments. As described in sections III-B and III-C, the *GB1113* dataset was used to select and learn features and weights for the various geometric properties and MP-DTW. As the sets consist of RGB video with manually annotated 2D hand positions, this group of experiments does not incorporate any 3D information from the Kinect. 3D trajectory matching and ICVM for 3D gesture properties remains future work.

We used two datasets from [25] as test sets, both of which are of fluent signers and comprise a combination of 1-handed and 2-handed signs of varying complexity. The *JK850* dataset consists of 850 unique ASL signs, while *CK368* contains 368 unique signs and is a more difficult set due to the wide variation in size and the exaggerated performance of signs compared to the models. The combined accuracy across both datasets is reported.

We compare our method to HMMs and LSTM networks. The complexity of the models is relatively limited due to the lack of training examples. For user independent experiments, the model parameters are selected based on performance on the validation set. The results reported are against the final testing set of *JK850* and *CK368*. For the LSTM network, a single layer is used before the LogSoftMax output layer, and the number of nodes is chosen based on validation performance. 190 and 320 nodes are used for one handed and two handed sign models, respectively. The network is trained with stochastic gradient descent and early stopping. Even with regularization techniques, the models were quick to overfit the data. For HMMs, a separate model is trained for each class. For 1-handed signs, best performance came from using a single state in each model. For 2-handed signs, each model contained 4 states. For both cases, the observation model is a single Gaussian. Final evaluation is performed using one-vs-all classification.

For the set of experiments, we define the measure of accuracy to be the percentage of signs whose correct match is ranked in the top k most similar signs for each $k \in \{1, 5, 10, 20, 30, 50, 100\}$. User-dependent experiment results are found in tables I and II, while user-independent results are contained in tables III and IV. The best performance in each table is emphasized in bold typeface.

A. User Dependent Experiments

User-dependent sign recognition can be useful in a system that learns from the user over time, as does the Nuance Dragon NaturallySpeaking voice dictation software [26]. As one uses a system based on our proposed methods, a learning algorithm can adapt the Gaussian models and ICVM and MP-DTW score weights to better fit that particular individual. For these experiments, the properties were selected and the weights trained using the test sets themselves, providing a measure of method potential and setting a goal for accuracy when training with different signers. Table I shows the results on the combined *JK850* and *CK368* datasets using manual annotations. We achieve a 9.2 percentage point increase in accuracy for rank $k \leq 10$.

TABLE I. User-Dependent Accuracy: Manual Annotations

	Maximum Rank k of Correct Match						
	1	5	10	20	30	50	100
HMM	0.206	0.452	0.557	0.652	0.719	0.785	0.870
LSTM	0.165	0.415	0.552	0.668	0.730	0.808	0.890
DTW	0.293	0.558	0.685	0.793	0.839	0.889	0.947
DTW + ICVM	0.316	0.597	0.730	0.821	0.852	0.897	0.947
MP-DTW	0.317	0.628	0.755	0.844	0.874	0.913	0.951
MP-DTW + ICVM	**0.333**	**0.646**	**0.777**	**0.855**	**0.878**	**0.918**	**0.955**

Since it is not realistic to expect a gesture recognition system to have access to manually provided annotations of hand locations, we performed the same experiments using the hand positions from the Kinect skeleton detector. Table II shows a significant increase in accuracy using these noisy data. As an example, recognition at rank $k \leq 10$ increases from 45.7% to 54.3% when using the automatic annotations.

TABLE II. User-Dependent Accuracy: Kinect Annotations

	Maximum Rank k of Correct Match						
	1	5	10	20	30	50	100
HMM	0.100	0.259	0.326	0.401	0.462	0.551	0.684
LSTM	0.089	0.246	0.335	0.444	0.516	0.616	0.743
DTW	0.162	0.372	0.458	0.562	0.607	0.682	0.791
DTW + ICVM	0.173	0.408	0.530	0.602	0.646	0.715	0.816
MP-DTW	0.199	0.399	0.493	0.596	**0.669**	0.702	0.820
MP-DTW + ICVM	**0.204**	**0.431**	**0.543**	**0.622**	0.664	**0.730**	**0.827**

B. User-Independent Experiments

The user-independent experiments demonstrate the potential improvement these methods provide in a pre-trained gesture recognition system that does not learn from the user. Table III shows the accuracy when using manual annotations. As can be seen, the best results again come from the combination of ICVM and MP-DTW. Table IV provides results using the Kinect skeletal annotations. It shows an increase in accuracy from 45.8% to 49.5% for maximum rank $k = 10$.

TABLE III. User-Independent Accuracy: Manual Annotations

	Maximum Rank k of Correct Match						
	1	5	10	20	30	50	100
HMM	0.158	0.359	0.459	0.582	0.647	0.725	0.831
LSTM	0.124	0.315	0.428	0.573	0.648	0.735	0.833
DTW	0.293	0.558	0.685	0.793	0.839	0.889	0.947
DTW + ICVM	0.298	0.590	0.714	0.802	0.849	0.897	0.950
MP-DTW	**0.336**	0.621	0.728	0.823	0.865	**0.908**	0.948
MP-DTW + ICVM	0.314	**0.625**	**0.731**	**0.824**	**0.867**	**0.908**	**0.957**

TABLE IV. USER-INDEPENDENT ACCURACY: KINECT ANNOTATIONS

	Maximum Rank k of Correct Match						
	1	5	10	20	30	50	100
HMM	0.089	0.220	0.296	0.376	0.435	0.512	0.645
LSTM	0.060	0.201	0.282	0.397	0.476	0.572	0.701
DTW	0.162	0.372	0.458	0.562	0.607	0.682	0.791
DTW + ICVM	0.176	**0.394**	**0.498**	0.576	0.637	0.713	0.815
MP-DTW	0.192	0.385	0.492	0.578	0.632	0.701	0.809
MP-DTW + ICVM	**0.197**	0.392	0.495	**0.592**	**0.650**	**0.727**	**0.823**

It is clear from these results that the best overall improvement comes from combining MP-DTW and ICVM features and that the two methods far outperform the HMM and LSTM network approaches in large vocabulary systems with few training examples per gesture class. In future work, we will be experimenting with other ways to use the ICVM features, including random decision forests, SVMs, a cascade filtering of potential matches, and a hierarchical classification system. Secondly, due to time required to do so, we did not train the individual MP-DTW passes with their own set of DTW score component weights $\{s_1, ..., s_6\}$ as discussed in section III-A and instead left it for future work.

V. CONCLUSION

In this paper, we proposed leveraging intra-class variation modeling and using multiple DTW passes, each focusing on a different gesture property and resizing technique, to improve exemplar-based gesture recognition in large vocabulary systems with a limited training set size. We demonstrated that either of the proposed methods alone improves recognition accuracy and, when combined, provide a significant increase in accuracy using both manually annotated hand positions and those from Kinect skeleton detectors.

ACKNOWLEDGMENT

This work was partially supported by National Science Foundation grants IIS-1055062 and CNS-1338118.

REFERENCES

[1] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state markov chains," *Ann. Math. Statist.*, vol. 37, no. 6, pp. 1554–1563, 12 1966.

[2] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 282–289.

[3] J. B. Kruskal and M. Liberman, "The symmetric time warping problem: From continuous to discrete," in *Time Warps*. Addison-Wesley, 1983.

[4] T. J. Darrell, I. A. Essa, and A. P. Pentland, "Task-specific gesture analysis in real-time using interpolated views," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 12, pp. 1236–1242, Dec 1996.

[5] A. Corradini, "Dynamic time warping for off-line recognition of a small gesture vocabulary," in *Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, 2001. Proceedings. IEEE ICCV Workshop on*, 2001, pp. 82–89.

[6] H. Wang, A. Stefan, S. Moradi, V. Athitsos, C. Neidle, and F. Kamangar, "A system for large vocabulary sign search," in *Proceedings of the 11th European conference on Trends and Topics in Computer Vision - Volume Part I*, ser. ECCV'10. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 342–353.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[8] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[9] A. Graves, "Supervised sequence labelling," in *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer Berlin Heidelberg, 2012, pp. 5–13.

[10] Y. Wang, D. Tran, Z. Liao, and D. Forsyth, "Discriminative hierarchical part-based models for human parsing and action recognition," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 3075–3102, Oct. 2012.

[11] Y. Tian, R. Sukthankar, and M. Shah, "Spatiotemporal deformable part models for action detection," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, June 2013, pp. 2642–2649.

[12] Y. Song, L. P. Morency, and R. Davis, "Action recognition by hierarchical sequence summarization," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, June 2013, pp. 3562–3569.

[13] B. Fernando, E. Gavves, J. O. M., A. Ghodrati, and T. Tuytelaars, "Modeling video evolution for action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 2015, pp. 5378–5387.

[14] O. Alsharif, T. Ouyang, F. Beaufays, S. Zhai, T. Breuel, and J. Schalkwyk, "Long short term memory neural network for keyboard gesture decoding," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*, 2015, pp. 2076–2080.

[15] I. Guyon, V. Athitsos, P. Jangyodsuk, B. Hamner, and H. Escalante, "Chalearn gesture challenge: Design and first results," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, 2012, pp. 1–6.

[16] J. Wan, Q. Ruan, W. Li, and S. Deng, "One-shot learning gesture recognition from RGB-D data using bag of features," *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2549–2582, 2013.

[17] J. Konečný and M. Hagara, "One-shot-learning gesture recognition using HOG-HOF features," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2513–2532, 2014.

[18] F. Jiang, S. Zhang, S. Wu, Y. Gao, and D. Zhao, "Multi-layered gesture recognition with kinect," *J. Mach. Learn. Res.*, vol. 16, no. 1, pp. 227–254, Jan. 2015.

[19] T. Kim, G. Shakhnarovich, and K. Livescu, "Fingerspelling recognition with semi-markov conditional random fields," in *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, 2013, pp. 1521–1528.

[20] H. Cooper, E. Ong, N. Pugeault, and R. Bowden, "Sign language recognition using sub-units," *Journal of Machine Learning Research*, vol. 13, pp. 2205–2231, 2012.

[21] H. Wang, X. Chai, Y. Zhou, and X. Chen, "Fast sign language recognition benefited from low rank approximation," in *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, vol. 1, May 2015, pp. 1–6.

[22] M. Reyes, G. Domnguez, and S. Escalera, "Featureweighting in dynamic timewarping for gesture recognition in depth data," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, Nov 2011, pp. 1182–1188.

[23] A. Yao, L. V. Gool, and P. Kohli, "Gesture recognition portfolios for personalization," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 1923–1930.

[24] M. Á. Bautista, A. Hernández-Vela, V. Ponce-López, X. Perez-Sala, X. Baró, O. Pujol, C. Angulo, and S. Escalera, "Probability-based dynamic time warping for gesture recognition on RGB-D data," in *Advances in Depth Image Analysis and Applications - International Workshop, WDIA 2012, Tsukuba, Japan, November 11, 2012, Revised Selected and Invited Papers*, 2012, pp. 126–135.

[25] C. Conly, P. Doliotis, P. Jangyodsuk, R. Alonzo, and V. Athitsos, "Toward a 3d body part detection video dataset and hand tracking benchmark," in *Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments*, ser. PETRA '13. New York, NY, USA: ACM, 2013, pp. 2:1–2:6.

[26] "Dragon 13 NaturallySpeaking home." [Online]. Available: http://www.nuance.com/dragon/index.htm