Single-Frame Hand Gesture Recognition Using Color and Depth Kernel Descriptors

Xiaolong Zhu and Kwan-Yee K. Wong
Department of Computer Science,
The University of Hong Kong,
Pokfulam Road, Hong Kong.
xlzhu@cs.hku.hk, kykwong@cs.hku.hk

Abstract

This paper presents a flexible method for single-frame hand gesture recognition by fusing information from color and depth images. Existing methods usually focus on designing intuitive features for color and depth images. On the contrary, our method first extracts common patch-level features, and fuses them by means of kernel descriptors. Linear SVM is then adopted to predict the class label efficiently. In our experiments on two American Sign Language (ASL) datasets, we demonstrate that our approach recognizes each sign accurately with only a small number of training samples, and is robust to the change of distance between the hand and the camera.

1 Introduction

Hand gesture recognition plays an important role in human machine interfaces, and it has a wide range of applications, including sign language learning, entertainment, gestural communication, non-intrusive motion capture system, *etc*. Traditional vision-based methods recognize hand gestures from color image sequence based on different features, such as edges, contours and textures. However, their performances depend greatly on how well they can segment the hand from the image. Much effort has therefore been put in locating and tracking the hand in the image sequence.

Recently, depth cameras become popular at a commodity price, and they bring us a new modality of data. Depth information provided by depth cameras makes the task of separating the hand from the background much easier. This makes it possible to obtain a reliable bounding box of the hand regardless of light and distance changes.

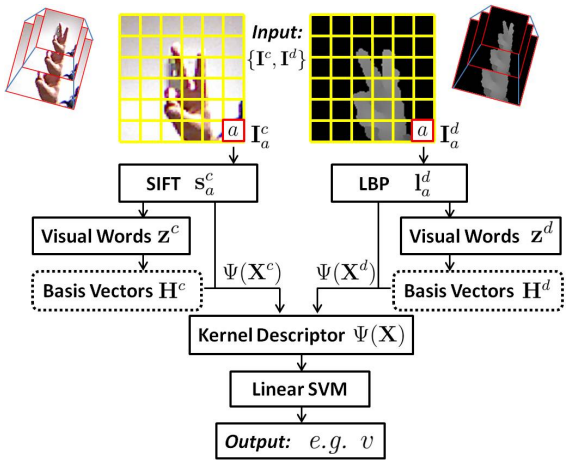

Figure 1. Work flow of our method.

In this paper, given the bounding box of the hand, we consider single-frame hand gesture recognition simply as an object recognition problem. We believe that local information can be well preserved in local patch descriptors. For example, patches in depth image describe local shape changes, whereas patches in color image capture appearance information such as texture and edge. As shown in Figure 1, we use kernel descriptor [3] to combine these features, and linear SVM classifiers to classify the images efficiently. Through experiments on two datasets, we demonstrate that our approach recognizes each sign accurately with only a small number of training samples, and is robust to the change of distance between the hand and the camera.

2 Related Work

Traditional vision-based methods attempt to recognize hand gestures from a color image. In general, they

can be classified into two major approaches, namely model-based approach and appearance-based approach.

In model-based approach, a generic hand model is usually created to track the hand before recognizing the hand gesture. Bjorn et al. [12] organized the hand configuration space into a hierarchical tree-based templates, and used Bayesian filters to locate them. However, their method requires the setting of many shape parameters during initialization of the hand model. In appearance-based approach, a number of hand images are first either rendered from synthetic model or acquired from a color camera. In the testing phase, prediction is made by a trained model from various clues like contour [1], edges [2], textures [6], *etc*. Note that hand detection is critical in almost all of the vision-based methods. When background becomes more complex, more effort is needed to differentiate the hand from the background.

Recently, the advent of depth cameras has raised great interests among computer vision community, and these cameras have been successfully used in many applications, e.g. pose estimation [7, 11], tracking [9], object recognition [3], *etc*. Not surprisingly, depth cameras are as well introduced to hand gesture recognition [13, 10]. Uebersax and Bergh [13] detected the palm in a depth image, applied three classifiers and finally aggregated the predictions into letter probabilities. Pugeault and Bowden [10] extracted features by applying Garbor filters on depth and color images, and used Random Forest to predict letter likelihood. Most of such methods design the features in an intuitive way, and then apply different classifiers to these features. How to design and fuse these features remains an open problem.

As a matter of fact, there have been efficient algorithms based on a depth image [11] that can track hand robustly. Hence it becomes possible to obtain a reliable bounding box of the hand. Meanwhile, the success of object recognition [3] and neural network [5] suggests that patch-based recognition is more natural and biologically plausible. *It is desirable to see if such patches-based algorithms are also applicable to hand gesture recognition.* We hereby propose to recognize different hand gestures using an efficient kernel matching [4], which can easily integrate patch-level features from different data modality, i.e. color and depth in this work, into image features by defining different kernel descriptors.

3 Kernel Descriptor for Linear SVM

There are two major concerns about applying patch-based algorithms to hand gesture recognition problem. First, appearance and shape of a gesture may vary from different users. Second, many clues may contribute to the recognition rate. In response to these concerns, an efficient approximation [4] is described in Sec 3.1 to deal with the appearance and shape variance, and kernel descriptors of color and depth images are introduced in Sec 3.2 for feature fusion.

3.1 Efficient Match Kernels for SVM

Let $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_m\}$ be a set of m patch features for an image $\mathbf{I}$. The match kernel of two images $\mathbf{I}_i$ and $\mathbf{I}_j$ for kernel SVM can be written as

$$K(\mathbf{X}_i, \mathbf{X}_j) = \frac{1}{|\mathbf{X}_i||\mathbf{X}_j|} \sum_{\mathbf{a}\in\mathbf{X}_i} \sum_{\mathbf{b}\in\mathbf{X}_j} k(\mathbf{a}, \mathbf{b}), \quad (1)$$

where each patch feature in $\mathbf{X}_i$ is compared with every patch feature in $\mathbf{X}_j$ by calculating the kernel function $k(\mathbf{a}, \mathbf{b}) = \phi(\mathbf{a})^\top \phi(\mathbf{b})$, and $\phi(\mathbf{a})$ and $\phi(\mathbf{b})$ are the kernel feature vectors. And note that it takes $O(N^2m^2)$ and $O(N^2)$ to compute and store the kernel matrix for kernel SVM, and this becomes infeasible when N, the number of training samples, grows larger. Therefore, the infinite-dimensional kernel feature vector $\phi(\mathbf{a})$ is approximated by a low D-dimensional vector $\psi(\mathbf{a}) = \mathbf{H}\mathbf{v}_\mathbf{a}$, where $\mathbf{H} = [\phi(\mathbf{z}_1), ..., \phi(\mathbf{z}_D)]$ is a collection of basis vectors in kernel feature space. We construct $\mathbf{H}$ by first extracting a set of visual words $\mathbf{Z} = \{\mathbf{z}_1, ..., \mathbf{z}_D\}$ using k-means algorithm. The coefficient vector $\mathbf{v}_\mathbf{a}$ can then be obtained by solving the following linear least squares problem

$$\mathbf{v}_\mathbf{a}^\star = \arg\min_{\mathbf{v}_\mathbf{a}} ||\phi(\mathbf{a}) - \mathbf{H}\mathbf{v}_\mathbf{a}||^2, \quad (2)$$

which has a closed-form solution $(\mathbf{H}^\top\mathbf{H})^{-1}\mathbf{H}^\top\phi(\mathbf{a})$. Hence, $k(\mathbf{a}, \mathbf{b})$ can be re-written as,

$$\begin{aligned} k(\mathbf{a}, \mathbf{b}) &\doteq \psi(\mathbf{a})^\top \psi(\mathbf{b}) \\ &= (\mathbf{H}\mathbf{v}_\mathbf{a}^\star)^\top \mathbf{H}\mathbf{v}_\mathbf{b}^\star \\ &= (\mathbf{H}^\top\phi(\mathbf{a}))^\top \cdot (\mathbf{H}^\top\mathbf{H})^{-1} \cdot (\mathbf{H}^\top\phi(\mathbf{b})) \\ &= \mathbf{k}_\mathbf{Z}(\mathbf{a})^\top \cdot \mathbf{K}_\mathbf{ZZ}^{-1} \cdot \mathbf{k}_\mathbf{Z}(\mathbf{b}) \end{aligned}$$

where $\mathbf{k}_\mathbf{Z}(\cdot) = [k(\mathbf{z}_1, \cdot), ..., k(\mathbf{z}_D, \cdot)]^\top$, and $\mathbf{K}_\mathbf{ZZ}^{-1} = (\mathbf{H}^\top\mathbf{H})^{-1}$ is the inverse kernel matrix of visual words. $\mathbf{K}_\mathbf{ZZ}^{-1}$ is positive definite, and thus it can be decomposed into $\mathbf{Q}^\top\mathbf{Q}$ by Cholesky Decomposition. Since $\mathbf{K}_\mathbf{ZZ}^{-1}$ only depends on visual words, $\mathbf{Q}$ can be learned from training data and calculated beforehand.

As a result, the original match kernel $K(\mathbf{X}_i, \mathbf{X}_j)$ can be simplified to $K(\mathbf{X}_i, \mathbf{X}_j) \doteq \Psi(\mathbf{X}_i)^\top\Psi(\mathbf{X}_j)$, where $\Psi(\mathbf{X}_i)$ and $\Psi(\mathbf{X}_j)$ are image-level features given by

$$\Psi(\mathbf{X}) = \frac{1}{|\mathbf{X}|} \sum_{\mathbf{a}\in\mathbf{X}} \mathbf{Q}\mathbf{k}_\mathbf{Z}(\mathbf{a}). \quad (3)$$

Intuitively, this approximation calculates the similarities between each patch and the visual words, and weights them in terms of the visual words by such similarities. Therefore, this descriptor is robust to small variance of a patch due to this soft-assignment scheme. On the other hand, it is common that different users may have slightly different appearances or hand shapes of the same hand gesture. Therefore, this kernel approximation is suitable for different users.

3.2 Color and Depth Kernel Descriptors

SIFT [8] and LBP [14] are the most commonly used patch-based features for human detection and object recognition. As shown in Figure 1, given a pair of color image $\mathbf{I}^c$ and depth image $\mathbf{I}^d$, we calculate the SIFT descriptor $\mathbf{s}_a^c$ for every patch a in $\mathbf{I}^c$ to encode gradient and edge information, and LBP descriptor $\mathbf{l}_a^d$ for every patch a in $\mathbf{I}^d$ to encode local shape changes. Visual words for color and depth patch descriptors are extracted separately from training set, and the kernel functions are defined as

$$\begin{aligned} k_{SIFT}(\mathbf{a},\mathbf{b}) &= \mathcal{G}(d(\mathbf{s}_a^c,\mathbf{s}_b^c)) \\ k_{LBP}(\mathbf{a},\mathbf{b}) &= \mathcal{G}(d(\mathbf{l}_a^d,\mathbf{l}_b^d)) \end{aligned} \tag{4}$$

where $\mathcal{G}(\cdot)$ is a Gaussian function and $d(\cdot,\cdot)$ is a distance function. The kernel descriptor $\Psi(\mathbf{X}^c)$ for color images and $\Psi(\mathbf{X}^d)$ for depth images are obtained by summing up all the local patch-level features according to (3). They are then concatenated to form an image-level kernel feature vector $\Psi(\mathbf{X}) = [\Psi(\mathbf{X}^c)\Psi(\mathbf{X}^d)]^{\mathrm{T}}$.

Image Pyramid Representation Image pyramid is used to strengthen the spatial support. As the kernel $K(\cdot,\cdot)$ compares two sets of features in the same region of two images, it can be seamlessly extended to a kernel function of image pyramid to embed spatial information of the patches in an image.

4 Experimental Results

We have evaluated our method on two datasets: ASL FingerSpelling Dataset [10], and our own dataset. In both datasets, we first randomly sampled a fixed number of image pairs, then trained 1-vs-all linear SVM for each class, and left the rest for testing. 1000 visual words were learned as basis vectors for $\mathbf{H}^c$ and $\mathbf{H}^d$ respectively. We repeated this process 5 times to obtain the final statistics.

In ASL FingerSpelling Dataset, 500 color images for each of 5 users are obtained for each sign. Since the signs 'j' and 'z' involve motion, they are not included in this experiment. We only sampled 40 images for each sign regardless of users and left the rest (2,460 images per class) for testing. The result is shown in Table 1. We outperformed the baseline approach [10] by 3%, without using image pyramid (NP). The result can be further improved to 12% using a 3-layer image pyramid (IP).

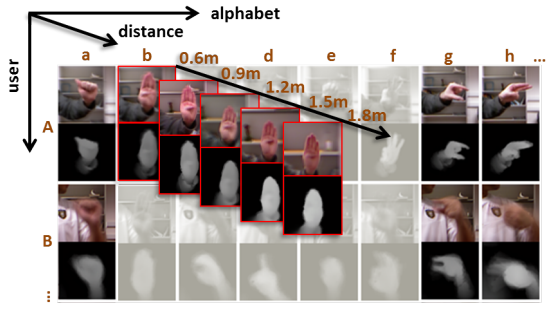

Figure 2. Our dataset. It consists of 24 static signs of 5 users at 5 different distances. The average images of each sign for each user at one distance are shown partially.

Method	#training samples	Overall Acc.
Pugeault [10]	1250	75 %
Our Approach (NP)	40	77.39 ± 0.13 %
Our Approach (IP)	40	88.94 ± 0.39 %

Table 1. Comparison with other method

Besides the difference between users, we also investigated another factor that we think is critical for recognition, i.e., the distance between hand and camera. We constructed a new dataset using Kinect by acquiring aligned depth and color image pairs from 5 users at 5 different distances from 0.6 meter to 1.8 meters, which are the typical working distances for desktop applications and home entertainment. The subjects were asked to make the sign facing the Kinect device, and the hand was tracked using off-the-shelf framework[1]. The ratio of the hand size relative to the size of image was fixed to 0.5, and each color or depth image was resized to 128×128. Some samples of this dataset are shown in Figure 2.

In this case, we randomly sampled 10% of the image pairs for each class, and trained four classifiers for color-only (CO), depth-only (DO), combined (ALL) features and combined features without image pyramid

[1] OpenNI. http://www.openni.org.

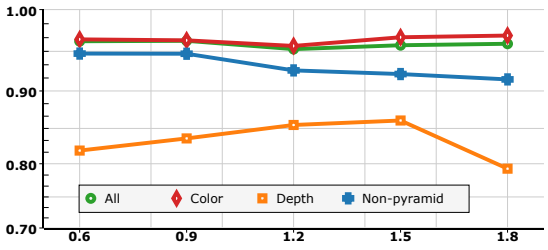

Figure 3. Overall accuracy w.r.t. distance between hand and camera.

(NP) kernel at different distances. The overall accuracy w.r.t. distance is plotted in Figure 3. In general, the combined color and depth kernel descriptor gives a high accuracy at all distances. Compared to depth descriptor (DO), color descriptor (CO) is more discriminative. When distance become longer than 1.5 meters, the quality of depth images of the hand decreases dramatically due to limited resolution, and this makes depth descriptor unreliable. The performance of combined features (ALL) therefore gets worse when the distance is long. The kernel of image pyramid (ALL) outperforms the kernel of single layer (NP), which suggests that spatial information of the image is helpful. We also show the overall confusion matrix in Figure 4. Despite over 90% overall accuracy, it is worth noting that it is still challenging for the signs, *m, n, s*, and *t* to be recognized, because these signs only differ by the thumb position.

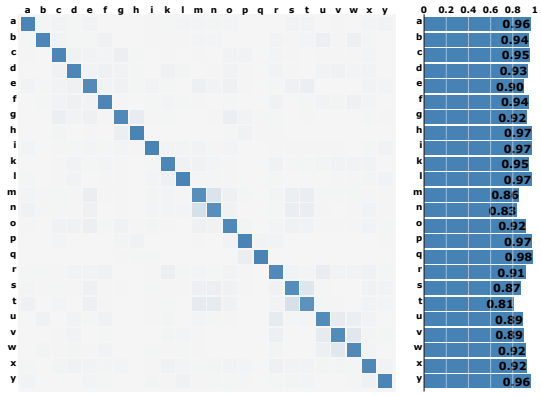

Figure 4. Confusion matrix and average accuracy.

5 Conclusion

In this paper, we apply a kernel descriptor for single-frame hand gesture recognition, which fuses color and depth information. As hand tracking is ensured using off-the-shelf method, we use patch-based features to recognize gestures. Our method outperformed intuitively hand-crafted features to a large extent and we believe that more static gestures can be added to this framework. In the meantime, we provide a dataset of different users at different distances for further comparison, and show that our approach does not depend on the distance between the hand and the camera.

References

[1] V. Athitsos and S. Sclaroff. An Appearance-Based Framework for 3D Hand Shape Classification and Camera Viewpoint Estimation. In *FG*, pages 40–45, 2002.

[2] V. Athitsos and S. Sclaroff. Estimating 3D hand pose from a cluttered image. In *CVPR*, pages II–432–9, 2003.

[3] L. Bo and D. Fox. Kernel Descriptors for Visual Recognition. In *NIPS*, pages 1–9, 2010.

[4] L. Bo and C. Sminchisescu. Efficient Match Kernels between Sets of Features for Visual Recognition. In *NIPS*, pages 1–9, 2009.

[5] D. C. Ciresan, U. Meier, and J. Schmidhuber. Multi-column Deep Neural Networks for Image Classification. In *CVPR*, 2012.

[6] M. de La Gorce, N. Paragios, and D. J. Fleet. Model-Based Hand Tracking with Texture, Shading and Self-occlusions. In *CVPR*, pages 1–8, 2008.

[7] G. Fanelli, J. Gall, and L. V. Gool. Real Time Head Pose Estimation with Random Regression Forests. In *CVPR*, pages 617–624, 2011.

[8] D. Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1150–1157, 1999.

[9] I. Oikonomidis, N. Kyriazis, and A. Argyros. Efficient model-based 3D tracking of hand articulations using Kinect. In *BMVC*, pages 101.1–101.11, 2011.

[10] N. Pugeault and R. Bowden. Spelling It Out : Real-Time ASL Fingerspelling Recognition. In *Proc. ICCV 2011 Workshop on Consumer Depth Cameras for Computer Vision*, 2011.

[11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-Time Human Pose Recognition in Parts from a Single Depth Image. In *CVPR*, pages 1297–1304, 2011.

[12] B. Stenger, A. Thayananthan, P. H. S. Torr, and R. Cipolla. Model-based hand tracking using a hierarchical Bayesian filter. *IEEE PAMI*, 28(9):1372–84, Sept. 2006.

[13] D. Uebersax, J. Gall, M. V. den Bergh, and L. V. Gool. Real-time Sign Language Letter and Word Recognition from Depth Data. In *ICCV*, pages 1–8, 2011.

[14] X. Wang, T. X. Han, and S. Yan. An HOG-LBP Human Detector with Partial Occlusion Handling. In *ICCV*, pages 32–39, 2009.