A Unified Framework for Concurrent Usage of Hand Gesture, Shape and Pose

Cem Keskin[1,2], Eray Berger[2] and Lale Akarun[1]
[1]Boğaziçi University Computer Engineering Department, Istanbul, Turkey
[2]Sigma Resarch and Development, Istanbul, Turkey
cem.keskin@cmpe.boun.edu.tr, eray@sigmard.com, akarun@boun.edu.tr

Abstract

This proposal introduces a framework designed to allow concurrent usage of different modalities regarding hand based interaction. In particular, hand gestures and shapes can be classified, and the hand pose, i.e. the hand skeleton can be estimated by the framework. Hand gestures are modeled with mixtures of hidden Markov models (mHMM) using spectral clustering. To classify hand shapes and to estimate the hand skeleton, we use randomized decision forests (RDF). Classification trees are employed in a discriminative setting to label hand shapes, and an object recognition by parts approach is used to fit a skeleton to the hand. Regression trees are used to map certain hand poses directly to the global orientation of the hand. The accuracy and the real time efficiency of the framework is demonstrated in a third party game, in which the user can perform combinations of hand gestures and shapes to navigate and control an avatar.

1. Introduction

Hand gestures are a natural part of human interaction, and a good candidate for human computer interaction, since they play a complementary role for speech and a primary role for sign languages. Vision based hand gesture recognition is a difficult problem to tackle due to the difficulty of locating and segmenting the hand, and due to the large inter–personal variations between the trajectories. However, with its ability to generate depth images, Kinect made the human body and hand detection and segmentation a simple task.

This proposal describes a unified framework for recognition of complex hand gestures that involve static or dynamic hand shapes, as well as hand motion. Hand shape and hand motion are different types of signals, and are usually modeled separately. While these are classification problems, since the output is a sequence or shape label, hand pose estimation is a regression problem, as it is used to infer a skeleton configuration from a continuous space. In this work, we propose to attack these three problems using separate models.

The paper is organized as follows: In Section 2, we describe the hand gesture classification method used in the framework. In Sections 4 and 3, the hand shape classification and hand pose estimation methods are mentioned. The framework that combines these methods is explained in Section 5. Finally, demonstration details are given in Section 6.

2. Hand Gesture Recognition

From a signal processing perspective, a hand gesture can be thought of as the output of a stochastic process. Therefore, the most common approach to tackle the problem has been through the use of graphical models such as the HMMs, conditional random fields (CRF), and their variants. The main difference between HMMs and CRFs is that HMMs are generative, whereas CRFs are discriminative. Discriminative models enjoy a better accuracy at classification tasks in general. However, introduction of a new class label with corresponding positive data has to be dealt with by re–training the entire model from scratch. On the other hand, if modeled with a generative model, each class has a separate associated model, and it is sufficient to train a new model for the introduced class. Therefore, HMMs are more appropriate for our case. For a detailed explanation and comparison of HMM and CRF variants, see [1].

HMMs with fixed probabilities are very simple graphical models, and therefore, they may not fit to the data well. An analogy can be drawn with the Gaussian distribution. Gaussian models are commonly preferred due to their simplicity and analytical elegance, yet they do not fit to the data that are not normally distributed. A standard ailment is to use a mixture model instead, which effectively clusters the data and models each cluster separately. The same idea is adopted to sequence classification tasks: As HMMs are simple models, a mixture of HMMs (mHMM) can be used to first cluster the sequences and then to model them.

Training an mHMM follows the same steps as in k–means clustering: Iteratively, each sample is assigned to the *closest* cluster, each cluster is modeled with the samples as-

sociated with it, and it is tested whether introducing a new cluster increases the likelihood of observing the data. In this work, we make use of dynamic time warping (DTW) and spectral clustering to initialize the clusters as in [2]. Then, we iteratively model each cluster with an HMM, reassign samples to the HMMs, and repeat these steps until no sample changes its cluster label in the reassignment phase. Each gesture class is modeled with a separate mHMM. The observations are a combination of hand shape class labels and the velocity vector.

3. Hand Pose Estimation

Hand pose estimation is the act of fitting a skeleton to the hand. There has been two major approaches to this problem lately. The first is a particle swarm optimization based method that effectively fits a model to the hand in near-real time, as proposed by Oikinomidis et al. [5], and the second one is the object recognition by parts approach by Keskin et al. , which uses RDFs to classify each pixel of the hand into a hand part label and estimates the joint locations [3]. We use the latter method in this work to estimate the hand skeleton. The details regarding this method can be found in [3].

4. Hand Shape Classification

Hand shape classification is the act of assigning a class label to an input hand image, representing a certain configuration of the hand. Inspired by the method described in [3], we formulated an RDF that directly recognizes hand shapes, in which every pixel votes for a hand shape label instead of a hand part label. The final class label is determined by majority vote. The details of the method can be found in [4].

5. The Unified Framework

The framework relies on Microsoft SDK to detect the approximate position of the hand. This can be done by simply using the body skeleton estimated by the SDK. Once the hands are located, a window around the hand is cropped, the mean depth of the hand is calculated using random samples around the hand centroid, and pixels with depths sufficiently far away are deleted. This segmented hand image is classified into a hand shape by using the trained RDFs as explained in Section 4. For certain outcomes, such as confident posteriors for a specific hand shape, we also use the same input image to estimate the hand pose. For instance, this method is used to detect a hand shape that imitates holding a joystick first, and then to fit a skeleton to it to estimate the parameters of the virtual joystick.

Hand gesture classification is active at all times. The framework estimates a set of posteriors for the hand shape label at each frame, and continuously uses these posteriors and the velocity vector as observations to spot and classify known gestures. While certain gestures make use of both motion and shape, there are gestures with pure motion and pure hand shape. These are determined simply by thresholding the magnitude of the velocity vector.

6. Demonstration

The framework is demonstrated in a third party game. The game features an avatar in a medieval setting, with actions such as moving around, crouching, jumping, attacking with swords or bows and casting spells. The framework described in this proposal will be used to capture the performed gestures by the user and to translate it into certain commands for the game. For instance, in the navigation mode, the user will be able to use his/her left hand to look around and his/her right hand to move by forming certain hand shapes and orienting them. In the spell casting mode, the user will be able to use combinations of motion and hand shape to perform spells: Drawing a circle in the air, and then forming a hand shape imitating a wolf will summon a wolf in the game; or imitating flames with fingers will select a fireball spell, and making a throwing gesture will launch it, etc. In the sword wielding mode, making an attacking gesture will thrust the sword, while a blocking gesture will raise the shield.

The framework is designed to run in the background and to send certain commands to the application running in the foreground. The methods are efficient, all of which are capable of running in real-time. The proposed framework can be used for other games, as well as other types of applications, such as browsers, multimedia players and modeling tools.

References

[1] C. Keskin, O. Aran, and L. Akarun. Hand Gesture Analysis. *Computer Analysis of Human Behavior*, pages 125–149, 2011. 1

[2] C. Keskin, T. Cemgil, and L. Akarun. Dtw based clustering to improve hand gesture. In *in Proceedings Human Behavior Understanding.HBU 2011*, pages 72–81, 2011. 2

[3] C. Keskin, F. Kirac, Y. E. Kara, and L. Akarun. Real-time hand pose estimation using depth sensors. In *Proceedings Thirteenth IEEE International Conference on Computer Vision Workshops. ICCV 2011*, pages 1228–1234. IEEE Comput. Soc, 2011. 2

[4] C. Keskin, F. Kirac, Y. E. Kara, and L. Akarun. Randomized decision forests for static and dynamic hand shape classification. In *Gesture Recognition Workshop in CVPR 2012*, 2012. 2

[5] I. Oikonomidis, N. Kyriazis, and A. Argyros. Markerless and efficient 26-DOF hand pose recovery. In *Proceedings of the 10th Asian conference on Computer vision-Volume Part III*, pages 744–757. Springer, 2011. 2