PerGrab : Adapting grabbing gesture recognition for personalized non-contact HCI

Tao Li and Ming Li

National Key Laboratory for Novel Software Technology
Nanjing University, Nanjing 210023, China
{lit,lim}@lamda.nju.edu.cn

Abstract. With recent development of technology, gesture has become a natural way to non-contact human computer interaction. In the literature, to improve user experience of this kind, there have been many works on gesture recognition. However, most existing works build a universal model for all users, which neglects the fact that different users may have different gesture styles. In this paper, rather than build an universal model for all users, we propose the PerGrab approach by building user-specific model for each user. It is expected the model can fit users' gesture well, hence leading to better performance. Specifically, given a universal model provided by manufacturers, PerGrab first records user-specific gesture styles by asking users to do some simple gestures, and then employs a personalization step to adapt universal model for the users. Experiments on applications show that PerGrab achieves good performance.

Keywords: Gesture recognition, transfer learning

1 Introduction

With the development of science and technology, more and more smart equipments appear in our daily life. Traditional touch operation interactions with these devices may be not appropriate in some environments, such as, touching devices are a common method of spreading infection in hospitals [1], large screen wall may be so large that some places are out of reach for touching operations. So, non-contact human computer interactions are needed. Naturally, we can use our gestures to communicate with the equipments. In recent years, gesture recognition has found its application ranging from medical rehabilitation to consumer electronics. In these applications, we can define different gestures for different commands. In the literature, many works [2, 3, 4] have been done.

However, most existing works build a universal model for all users, which neglect the fact that different persons may have different gesture styles. For example, Fig.1 shows the grab and release pictures, where (a) and (b) are from one person and (c) and (d) are from another. It is clearly that, these two persons hold different grab and release gesture styles. If a model build on the data does not cover some users' gesture styles, it will be difficult for such a model to achieve

good performance on the corresponding users' gestures. For example, some users can not communicate with smart TV well because the model offered by the manufacturer is not fit for them. However, it is impossible for the manufacturer to cover all the gesture styles. In consequent, building a model fit for every user is a very challenging task. A better way is that, we just need to collect a few users' data and use these data to personalize the model offered by the manufacturer. So, transfer learning can be used to address this problem. To the best of our knowledge, it is the first work to employ transfer learning in gesture recognition. For simplicity, we consider two gestures recognition problem for example, fist as "left-click"on the mouse and palm as "release left-click ".

(a) (b) (c) (d)

Fig. 1. different persons have different gesture styles

In this paper, we propose an efficient approach called PerGrab (Personalized Grab), which enables users to do a little work to get better performance in hand gesture recognition. In the first step, the users are asked to do some sample gestures, such that some user-specific labeled data can be collected. Then based on the collected data, we personalize the universal model offered by the manufacturer by transfer learning. Experiments on the real-world application of PerGrab show that, the performance of PerGrab is very good.

The rest of this paper is organized as follows. Section 2 introduces related work. Section 3 describes the PerGrab approach including the framework of our system and transfer learning method. In section 4, experiments and results are reported, which are followed by conclusion.

2 Related work

Vision based hand gesture recognition plays a very important role in human computer interaction (HCI) for it is a non-contact interaction method. In the previous vision based work [2, 5, 6], hands tracking and segmentation are challenging when users are in a complex background. With the development of inexpensive depth sensors, such as Kinect, tracking and segmenting become simpler. Recently, a lot of hand gesture recognition approaches are implemented based on Kincet [4, 7]. They build good performance recognition models for the applications, but they did not consider the fact that different persons may have different gesture styles.

The idea of transfer learning comes from that previous knowledge can help people learn similar new knowledge. Pan and Yang [8] shows transfer learning

finds its utilities in many applications, such as Web-document classification [9], sentiment classification [10], and indoor WiFi location [11, 12]. In these applications, there are only few labeled data for the learning task, while a lot of labeled data available for tasks which are related or similar to the problem. Transfer learning can achieve better performance via exploiting data from different but related tasks.

Due to the success of transfer learning in many applications, we can also treat different persons' hand gesture data as different domains and formulate the hand gesture recognition as a transfer learning problem to get better results.

3 The PerGrab approach

3.1 The general framework

The framework of our PerGrab approach is presented in Fig.2. Compared with the traditional universal model methods, PerGrab personalizes the model offered by the manufacturer using a few user's own data, if the user gesture style is different from the manufacturer's.

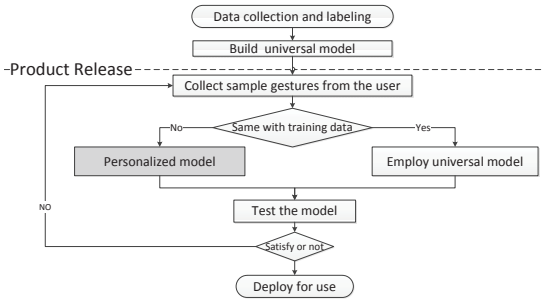


Fig. 2. The PerGrab framework

In the process of PerGrab, before product release, some hand gesture data are collected and labeled, and then a universal model is built. After product release, when users use their equipments for the first time, an initial setting is offered to them. They will be asked to do some sample gestures obeying the setting program to record some users' labeled data. After collecting a few users' data, non-parametric two-sample test Maximum Mean Discrepancy (MMD) [13] will be used to judge whether the customer's gesture style is similar to the training data offered by the manufacturer. If they are similar, the universal model will be directly employed, otherwise transfer learning method will be used to personalize manufacturer's universal model to get a user-specific one. Finally, the users can communicate with the equipment by their hand gestures on user-specific model.

3.2 Adapting gesture recognition model for personalization

As shown in Fig.2, the "Personalized model" step is the core of PerGrab. Let $x \in R^n$ denote a sample, $y \in \{+1, -1\}$ as the corresponding label. Due to different persons may have different hand gesture styles and the limitation that only a few users' data are available, this personalized model problem can be formulated as a transfer learning problem. Manufacturer offers plenty of labeled source domain data and users offer a few target domain data by initial setting. Therefore, we can personalize the manufacturer's universal model. Set $D_S = \{(x_{s_1}, y_{s_1}), (x_{s_2}, y_{s_2}), ..., (x_{s_{ns}}, y_{s_{ns}})\}$ as the data for source domain (ns is the source sample size). Moreover, $D_T = \{(x_{t_1}, y_{t_1}), (x_{t_2}, y_{t_2}), ..., (x_{t_{nt}}, y_{t_{nt}})\}$ donates data for target domain (nt is the target sample size).

Although different persons may have different gesture styles, their gestures should be similar, and parameters of each user's optimal recognition model are highly related. Specifically, we assume that parameters of each model can be spilt into two parts, one common part w_0 and the specific part w_t (t is the related model number). The problem is formulated as transfer learning, where manufacturer's plenty of labeled data are viewed as source domain and users' a few labeled data are recorded as target domain. Set w_s and w_t as source domain's parameters and target domain's parameters, which can be presented as follows:

$$w_s = w_0 + v_s \qquad and \qquad w_t = w_0 + v_t \tag{1}$$

where w_0 is the common part of the two domains, v_s is the source-specific part and v_t is the target-specific part.

We formulate the problem as follows:

$$\begin{aligned} \underset{w_0, v_t, \xi_{t_i}}{min} \ & \frac{1}{2}\|w_0\|^2 + \frac{C_1}{2}\|v_s\|^2 + \frac{C_2}{2}\|v_t\|^2 + C_3 \sum_{i=1}^{n_s} \xi_{s_i} + C_4 \sum_{i=1}^{n_t} \xi_{t_i} \\ s.t. \quad & y_{s_i}(w_0 + v_s) \cdot x_{s_i} \geq 1 - \xi_{s_i}, \xi_{s_i} \geq 0 \\ & y_{t_i}(w_0 + v_t) \cdot x_{t_i} \geq 1 - \xi_{t_i}, \xi_{t_i} \geq 0 \end{aligned} \tag{2}$$

The first three parts of equation (2) are regularization terms. The first term is common part parameters, the second is source-specific part and the third is target-specific part. We use C_1 and C_2 to adjust their relationship. The last two parts are the sum of hinge loss of source domain and target domain. Different from the multi-task learning [14] that care about all the tasks' performance, we just focus on the target domain's performance in the transfer learning. So we give a higher value to C_4 than C_3.

We rewrite the formulation above into the dual form:

$$\begin{aligned} \underset{\substack{0 \leq \lambda_s \leq C_3 \\ 0 \leq \lambda_t \leq C_4}}{\min} \ & \frac{1}{2} \begin{pmatrix} \lambda_s \\ \lambda_t \end{pmatrix}^T \begin{pmatrix} \left(1 + \frac{1}{C_1}\right) A_S & A_{ST} \\ A_{TS} & (1 + \frac{1}{C_2}) A_T \end{pmatrix} \begin{pmatrix} \lambda_S \\ \lambda_T \end{pmatrix} + \mathbf{1}^T \begin{pmatrix} \lambda_S \\ \lambda_T \end{pmatrix} \\ s.t. \quad & \begin{pmatrix} y_S^T & \mathbf{0} \\ \mathbf{0} & y_T^T \end{pmatrix} \begin{pmatrix} \lambda_S \\ \lambda_T \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} \end{aligned} \tag{3}$$

where C_1, C_2, C_3, C_4 are the same as equation (2). A_S is the kernel for the source domain, with element $A_{S_{i,j}} = K(x_{s_i}, x_{s_j}), \quad x_{s_i}, x_{s_j} \in D_S$. A_{ST} is the kernel for the source and target domain, with element $A_{ST_{i,j}} = K(x_{s_i}, x_{t_j}), x_{s_i} \in D_S, x_{t_j} \in D_T$. A_T and A_{TS} are similar. The kernel can be linear, polynomial or others. In this paper, we use the linear kernel. Mosek is employed to deal with this quadratic programming problem in seconds.

4 Experiments

There are 8 persons taking part in the experiment, and each person offers some data using the sampling program we design. The program tells each participant to do some grab and release gestures, and records his/her hand areas in 100 × 100 pixels. We record 2000 data for each person, 1000 positive (fists) and 1000 negative (palms).

Features used in this paper are Histogram oriented Gradient (HoG) in 4000 dimensions and Gist in 512 dimensions. The baseline methods are SVM and Random Forest, which are widely used in computer vision applications.

Three experiments are conducted in this section. Firstly, we prove different persons may have different gesture styles. Secondly, model personalization by adapting the universal can get good performance. Finally, we show PerGarb gets better performance when the number of training persons increases. Average F1 measure over 50 rounds is used for fist.

4.1 Experiment 1: universal model does not work for every user

We want to prove different persons may have different gesture styles. We think if the universal model does not cover someone's style, it will not work for him/her. we conduct experiments as follows. One person is assigned as training data, randomly sampling 1000 samples (500 fists + 500 palms) to train a model, then test the model on all the 8 persons' data and get 8 results. The testing data consist of 1000 samples (500 fists +500 palms), the training data and testing data are without overlapping when they come from the same person. We can get 8 groups of results by changing the training person.

Table 1. Average Fist F1 measure of each pair of classifiers and features

	SVM(Polynomial)	SVM(Linear)	SVM(RBF)	Random Forest
HoG	**.937 ± .047**	.932 ± .048	.932 ± .048	.907 ± .062
Gist	.916 ± .065	.915 ± .061	.921 ± .057	.912 ± .050

We conduct the experiments of all pairs of features and baseline methods with cross-validation to get the each best results. The average results are shown in Table 1. We find the pair of SVM with polynomial kernel and HoG feature gets the best performance, so we choose this pair on behalf to conduct our following

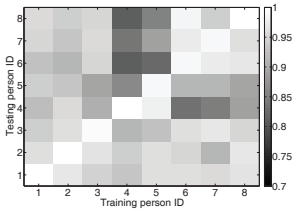


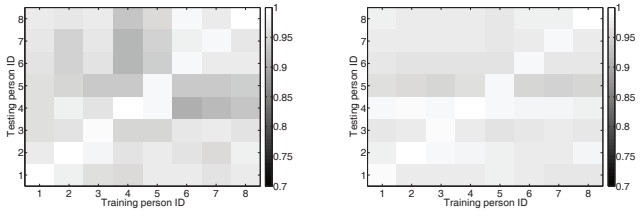
Fig. 3. Results of the model of SVM(Polynomial) + HoG

experiments. In details, the 8 × 8 result matrix of this pair of classifier and feature is shown in Fig.3, while the other pairs get the similar results.

Fig.3 shows the results of the model on HoG feature and SVM with polynomial kernel. We can see that each model trained by one person's data can get good performance on some persons, but get bad performance on some other persons' data. For example, when the training person ID = 3, the model performs good on Testing person ID = 6,7,8, but bad on Testing person ID = 4,5. The similar results appear when other pairs of features and classifiers are employed. So, It is clear that different persons may have different gesture styles, and a universal model can't satisfy all the users.

4.2 Experiment 2: adapting universal model helps

Each training person in the Experiment 1 is selected as source domain, with 950 samples (475 fists + 475 palms) sampled, and its testing persons are selected as target domains, offering 50 target domain samples (25 fists + 25 palms) and 1000 testing samples (500 fists + 500 palms). Parameters of PerGrab are set as $C_1 = C_2 = C_3 = 1$ and $C_4 = 10$. At the same time, we also simply combine the source and target data as the new training data under the unified distribution assumption and build the model using SVM. The two 8 × 8 result matrixes are shown in Fig.4.



(a) SVM (b) PerGrab

Fig. 4. The results of SVM and PerGrab trained by source and target domain data.

Fig.4 shows results of PerGrab and its compared method. After sign test by t-test with significance level 0.05, PerGrab gets 40 wins, 19 ties and 5 loses, which is significantly better than the results shown in Fig.3. From the results, we can say transfer learning really helps in the hand gesture recognition application.

4.3 Experiment 3: when more persons are added to training set

In this subsection, we conduct experiment in conditions where more source persons' data are available. The experiment is conducted with source domain person number ranging from 1 to 5. For the fair of the experiment, the sum of all the source data size and target data size is set at 1000. Assume we have N source persons' data, we can get N transfer models for each pair of source person and target person. And when target person's testing data come, each testing data will get N predict labels by N transfer models. Finally, all the predicted labels are ensembled by majority voting to get the final label prediction. The results are shown in Fig.5.

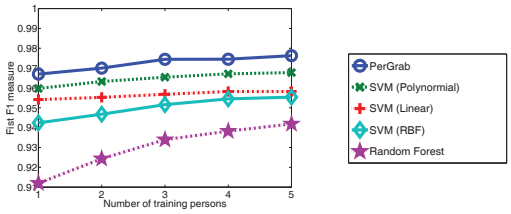


Fig. 5. The results of PerGrab and the comparison methods as the number of training persons increases

Fig.5 shows that as the number of training person increases, the performance of PerGrab and comparison methods become better. Moreover, our PerGrab gets the best performance among all the methods.

5 Conclusion

In this paper, we discuss the problem that different persons may have different gesture styles, due to which, universal model is not able to give satisfactory performance for all users. And we introduce the PerGrab approach, which adapts the universal model offered by manufacturer to the user-specific model. PerGrab just needs user to do a little work to collect a few data as user-specific data, which are used to personalize the model offered by the manufacturer. The experiments show PerGrab performs better than universal distribution formulation.

In the current paper, we only consider the grab and release gestures. The framework of PerGrab also can be applied to other gestures recognition applications, such as gestures in sign language. In the future, we will continue our research on other complex gestures recognition.

6 Acknowledgments

The work is supported by the National Science Foundation of China (No. 61272217), and National Social Science Funds of China (No. 11AZD121) and the 2013 State Grid Research Project.

References

[1] Schultz, M., Gill, J., Zubairi, S., Huber, R., Gordin, F.: Bacterial contamination of computer keyboards in a teaching hospital. Infection Control and Hospital Epidemiology **24**(4) (2003) 302–303

[2] Bretzner, L., Laptev, I., Lindeberg, T.: Hand gesture recognition using multi-scale color feature, hierarchical models and particle filtering. In: Proceedings of the 5th Face and Gesture. (2002) 423–428

[3] Zhang, X., Chen, X., Wang, W., Yang, J., Vuokko, L., Wang, K.: Hand gesture recognition and virtual game control based on 3d accelerometer and emg sensors. In: Proceedings of the 14th IUI. (2009) 401–406

[4] Ren, Z., Yuan, J., Zhang, Z.: Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In: Proceedings of the 19th ACM MM. (2011) 1093–1096

[5] Stenger, B., Thayananthan, A., Torr, P.H., Cipolla, R.C.: Filtering using a tree-based estimator. In: Proceedings of the 9th ICCV. (2003) 1063–1070

[6] Wang, C.C., Wang, K.C.: Hand posture recognition using adaboost with sift for human robot interaction. In Lee, S., Suh, I., Kim, M., eds.: Recent Progress in Robotics: Viable Robotic Service to Human. Springer (2008) 317–329

[7] Li, H., Yang, L., Wu, X., Xu, S., Wang, Y.: Static hand gesture recognition based on hog with kinect. In: Proceedings of the 4th IHMSC. (2012) 271–273

[8] Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering **22**(10) (2010) 1345–1359

[9] Dai, W., Yang, Q., Xue, G., Yu, Y.: Boosting for transfer learning. In: Proceedings of the 24th ICML. (2007) 193–200

[10] Blitzer, J., Dredze, M., Pereira, F.: Biographies, bollywood, boomboxes and blenders : Domain adaptation for sentiment classification. In: Proceedings of the 45th ACL. (2007) 432–439

[11] Zheng, V., Pan, S.J., Yang, Q., Pan, J.: Transferring multi-device localization models using latent multi-task learning. In: Proceedings of the 23rd AAAI. (2008) 1427–1432

[12] Pan, S., Zheng, V.W., Yang, Q., Hu, D.H.: Transfer learning for wifi-based indoor localization. In: Proceedings of the 23th AAAI. (2008)

[13] Gretton, A., Borgwardt, K.M., Rasch, M., Schölkopf, B., Smola, A.: A kernel method for the two-sample-problem. In: Proceedings of the 19th NIPS. (2007) 513–520

[14] Evgeniou, T., Pontil, M.: Regularized multi-task learning. In: Proceedings of the 10th KDD. (2004) 109–117