Hand, head detection in first person view video

Hao Lu
School of Informatics and Computing
Indiana University Bloomington
Bloomington, IN
luha@indiana.edu

***Abstract*—In children word learning behavior study, knowing how the parent and child hands and face will effect children learning process is very import. This project is working on the children word learning video, use skin color filter to get the rough skin blobs. Then calculate the optical flow information for each two frames, based on the optical flow information and skin blobs, merge the blobs when those have very close high dimension distance. Based on the blobs temporal information or using Kalman filter to generate the blobs trajectory in video. With blobs trajectory active areas and blobs optical information, give blobs trajectory definition, whether it is face or hand. We present an inference method that can predict the best sequence of the hands and face associated action from an input sequence of images.**

Keywords—Graph cut; skin filter; optical flow; hand and head detection

I. INTRODUCTION

Faces are accessible 'windows' into the mechanisms that govern our emotional and social lives. The face is a unique feature of human beings. Even the faces of "identical twins" differ in some respects. Humans can detect and identify faces in a scene with little or no effort. This skill is quite robust, despite large changes in the visual stimulus due to viewing conditions, expression, aging, and distractions such as glass or changes in hair style.

Advances in camera miniaturization and mobile computing made the camera feasible to capture and process photos and videos from the camera worn on a person's body. So that made egocentric video is one of the hot computer vision and image processing topic recently. Current, there are a lot of products have this feature, such as Google glasses, Go-pro. So with this kind of equipment psychology research can have difference cue and simulate the human visual information to analyses those problem. Such like for the children word learning study or other human behavior study, psychology researcher really want to know during the new words learning process, how the children's both hands and face movement, how the teacher's hands effect the children's attention. So to know the face and hands location in the camera will be very informative for psychology research. My final project is trying to build algorithm, can categorize the people's hands and head in egocentric video, and also give the hands specific tag, like right hand or left hand and also whose. For the video which I will process, there will be two hands belong people who wear camera, two hands and face belong to the people faced with. Up to this time, only few work is focus on the first person view analyses, most of them work on activity recognition, objects recognition[1][2][3][4]. Currently, no group is working on hands and head recognition in this kind of data.

vision-based techniques for under-standing and recognizing ego-actions have focused largely on the use of outside looking cameras to capture information about the visual world, such as hand gestures, objects of interactions and ego-motion[4][7] . Recent work has also shown that a user's focus of attention on a rough macro-scale (i.e. head pose and detected objects or faces, ands) can be used to model social interactions[5]. Kitani et al. [6] demonstrated in their recent work that global ego-motion has been shown to be a successful descriptor for human action in sports.

II. GENERAL IDEAL

The experiment is record about 120 second video records by child head mounted camera. During the experiment, parent and child seat on two sides of the table, they both are wearing head mounted cameras, and motion sensor in the hand, the device, looks like a glove. Child plays the 3 toys all the time. Parent is trying to teach the child about 3 toys name.

Fig. 1. Image sample from experiment camera. Top left is the image taken by camera from back side of the child.Top right image is the image taken by camera from back side of the parent. Bottom left image is taken by the children head mounted camera. Bottom right image is taken by camera from parent head mounted camera.

Our goal is to model and detect primitive ego-action categories using first-person sensing. Recognizing primitive actions are important for understanding human activities because they can be used as building blocks to understand more complex high-level activities. This project have six steps.

- With high precision skin detection as seed, use graph cut generate the skin blobs for each frames.
- Based on the frame optical flow information give every skin blobs optical flow features.
- For each blob, based on the blobs distance, optical flow distance, KNN decide whether merge the blobs.
- Based on the blobs sequential info, generate the tube
- Tube filter, for the tube which length is no longer enough, filter it.
- Tube classify based on the common sense.

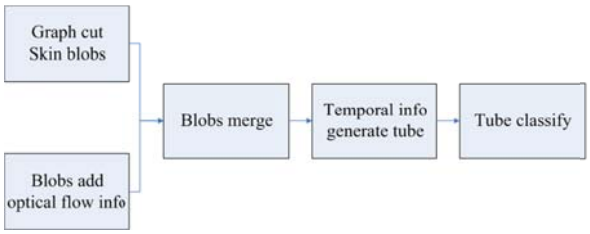

Fig. 2. The flow chart of the general method used in this project

A. *Graph cut for skin blobs*

Human skin segmentation is a research field with many applications such as video surveillance, face or hand gesture recognition, content-based visual information retrieval (CBVIR), filtering on the web and others. This is a fundamental task for any application that looks for human sequences on image and video streams. In many situations, very little human intervention must occur (as when the data to be processed is represented by a huge database). This project presents a new automatic per-pixel human skin segmentation method, which is highly customizable and yields good results compared with other relevant works.

Most skin segmentation methods are color based approaches and introduce a color metric which considers the distance of the pixel to a specific color [8][9]. It is not a trivial problem to solve, since objects and backgrounds exist in a large variety of colors, including skin tones. But for our experiment the color tone is very simple, few area region is close to the skin color. This project is based on the premise that skin colors form a small and unique subset of the RGB color space, which makes it easier to solve this specific case of segmentation. In this approach the segmentation is modeled as a min-cut problem on a direct graph [10][11] and each edge has a well-defined cost.

With very high precision skin detection as seed, we get a positive region of the image. It is possible to find a characteristic function of an object defined in a given domain by minimizing an objective function, i.e., given a set V, we have to find the characteristic function X which is the minimum argument of a function [10] and the partition sets. A widely used objective function in image segmentation is the Gibbs Energy[10] defined as

$$E(X) = \sum_{x_i \in V} E_1\big(X(x_i)\big) + \lambda \sum_{x_{i,x_j} \in \xi} E_2\big(X(x_i), X(x_j)\big) \quad (1)$$

Where x_i *and* x_j are element of the set to be segmented, V is the set of elements, ε is the set of connected elements and λ is a weight. E_1 is the term that defines the cost for each x_i to one of the sets. Aiming to minimize the objective function, the cost should be inversely proportional to the probability of x_i belonging to the set. It can be given as

$$\begin{array}{lll} E_1(X(x_i)=1)=0 & E_1(X(x_i)=0)=\infty & \forall x_i \in O \\ E_1(X(x_i)=1)=\infty & E_1(X(x_i)=0)=0 & \forall x_i \in B \\ E_1(X(x_i)=1)=\phi(\rho_a) & E_1(X(x_i)=0)=\phi(\rho_b) & \forall x_i \in N \end{array} \quad (2)$$

where O is the set of object elements, B is the set of the background elements, N is the set of pixels whose labels are unknown and ϕ is a function inversely proportional to its parameters terms ρ_a and ρ_b.

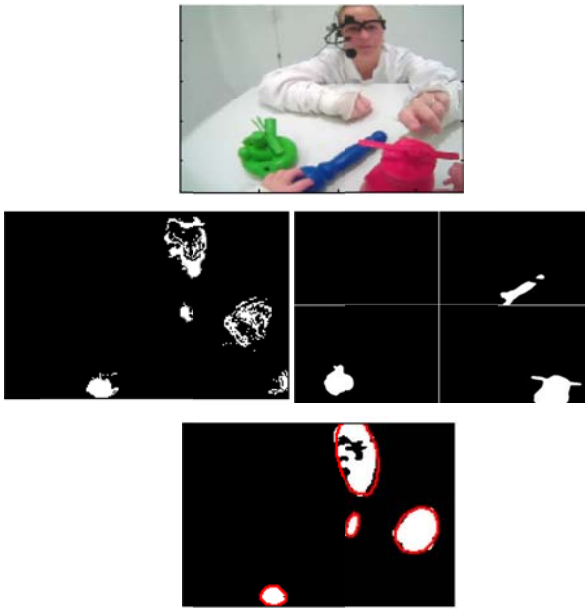

Fig. 3. Graph cut procudure. Top image is the image from the video. Second line images: left is the seed of the positive result which means that every pixel is have very high chance of the skin

region. Right is the seed of the oppsitive region. Bottom image is the graph cut result.

B. *Optical flow*

Optical flow is the approximated motion vector at each pixel location. It can tell us about the relative distances of objects, as closer moving objects will have more apparent motion than moving objects that are further away, given equal speed.

Optical flow is based on the assumptions that objects in the image at time *t* will generally still be in the image at time *t+1*, only they will be displaced. This is represented by Ce Liu *et.al*[12][13].

The sample result is shown as follow:

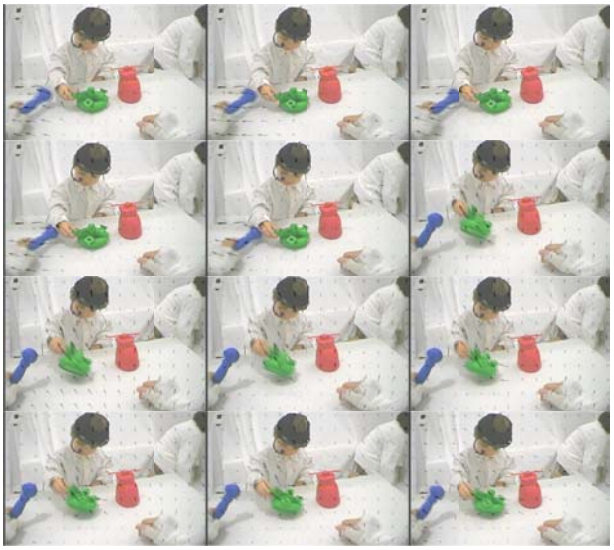

Fig. 4. Optical flow sample result.
the images showed above are have arrow shows the pixels value and the diction.

The optical flow information I will saved as two types of image, in order can make sure all the value in 0-255 scale. For value each pixel I saved as:

$$Vvalue = 40 * \log(\sqrt{Vx^2 + Vy^2} + 1)$$

For angle each pixel I saved as:

$$\text{Vangle} = \frac{255}{360}\text{atan}(\frac{Vy}{Vx})$$

C. *Get blobs optical flow info*

Based on the previous two steps, I generate the blobs optical flow information for each blob. So for each blob have 4 values: 1, blob pixels optical flow value mean, 2 blob pixels optical flow angle mean, 3 blob pixels optical flow value variance, 4 blob pixels optical flow angle variance.

D. *Merge close blobs (KNN)*

Cause child and parent wear gloves which can get the motion date during the experiment. So sometimes gloves cut the hand in to several parts in the camera view. So it was one very crucial problem in this vision problem.

For this problem we used KNN algorithm to classify the blobs which belong to one of the hands or head. The KNN method procedure in my project as follow:

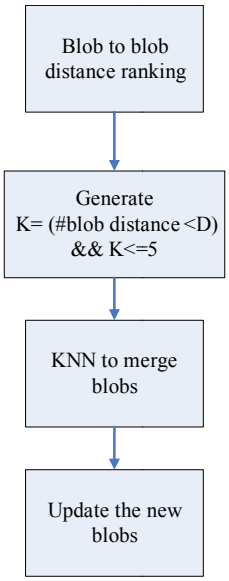

Fig. 5. KNN for merge the same topic blobs

Blobs will ranking based on the each blob to other blobs distance. The distance will calculate by:

$$blob2blob.dist = b2b.coord + a1 * b2b.opt\_ang\_vari + a1 * b2b.opt\_ang\_vari + a1 * b2b.opt\_ang\_vari + a1 * b2b.opt\_ang\_vari$$

If one frame have N blobs, each blob will have N-1 distance values, and ranking all for every blob. Then count the K value based on the how many distance large than D value. If have m distance values large than D, The K value is $round\left(\frac{m}{N} + 1\right)$. The previous condition is that K never larger than 5, that is because only have 4 hands and one face in this experiment.

So once we define the K value base on last step. We can based on the classic KNN algorithm do the merge blobs. Then still need to do the blob update. For update, I mean that new blobs will have new optical flow information and area, and central, area based on the blobs which will merge to it.

Here is the merge example:

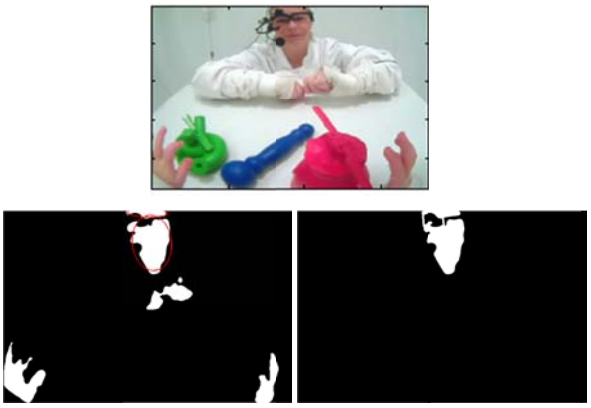

Fig. 6. Merge blobs example

Top image is the frame image data from the video. Cause parent wear the head camera with black belt that block parent head to two parts: forehead and face region. The bottom left is the blobs that recognize by the previous graph cut procedure. The bottom right is the blobs which based on the KNN algorithm merge together result.

In here, for K define method only give K as 4, so that means only two of the blobs will merge together. Compare with distance, the red eclipse region will merge together.

E. Generate the tube

In this step we solve the problem that we connect sequential frames' 2-D blobs based on the blobs location information. The reason is because for hand or for face blobs, those blobs move slowly if the frame rate is large that 30 Hz. Usually for same hand blobs, from one frame to next frame it will not move a lot and the hand optical flow information also in some close range.

Some method may also can used in this problem, like Kalman filter tracking the blobs. But for this experiment, hands and face always move in roughly same area on camera view. Kalman filter is not a good method to working on this problem. Especially, in some case two hand may blocked by each other, and they have the same color tone, it is highly chance that when two hands merge together it might reach dead end. Kalman filter may good at that even pace object tracking, or the block object wont move case.

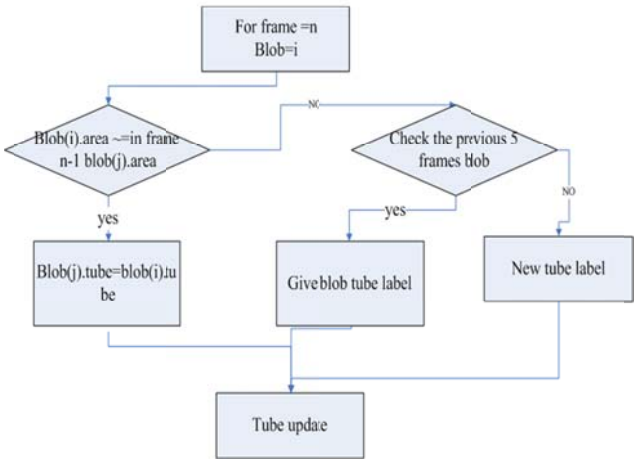

Fig. 7. Generate tube flow chart

Based on the temporal information, we generate about 300 tubes for 30s 1000 frames. As we can see in the Fig.8, most of the tube duration is less that 30 frames. It could caused by the previous skin blobs piece or skin move to camera view then move out very quickly. So I trade the duration low than 30 frame tube I filter them all.

After filter the short tube, only about 30 tubes left.

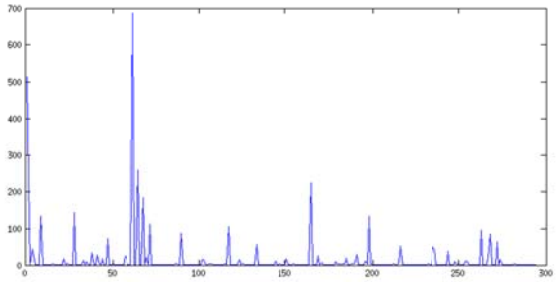

Fig. 8. 300 tubes duration

F. Tube categorize

Once we have about 30 long tubes, we based on the common sense to give the tubes definition. The common sense means, for other person's head will always in the top center of the view, other person 's hand also locate at the right and left of the view, between the head and self hands. The view should be looks like this.

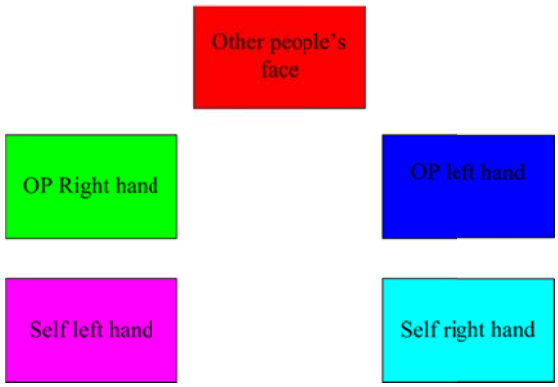

Fig. 9. Head, hands active region

So based on the common sense we can finally give each tube definition.

III. RESULT

I will attached result video. In order to prove the common sense classification about tube, I plot the blobs location information.

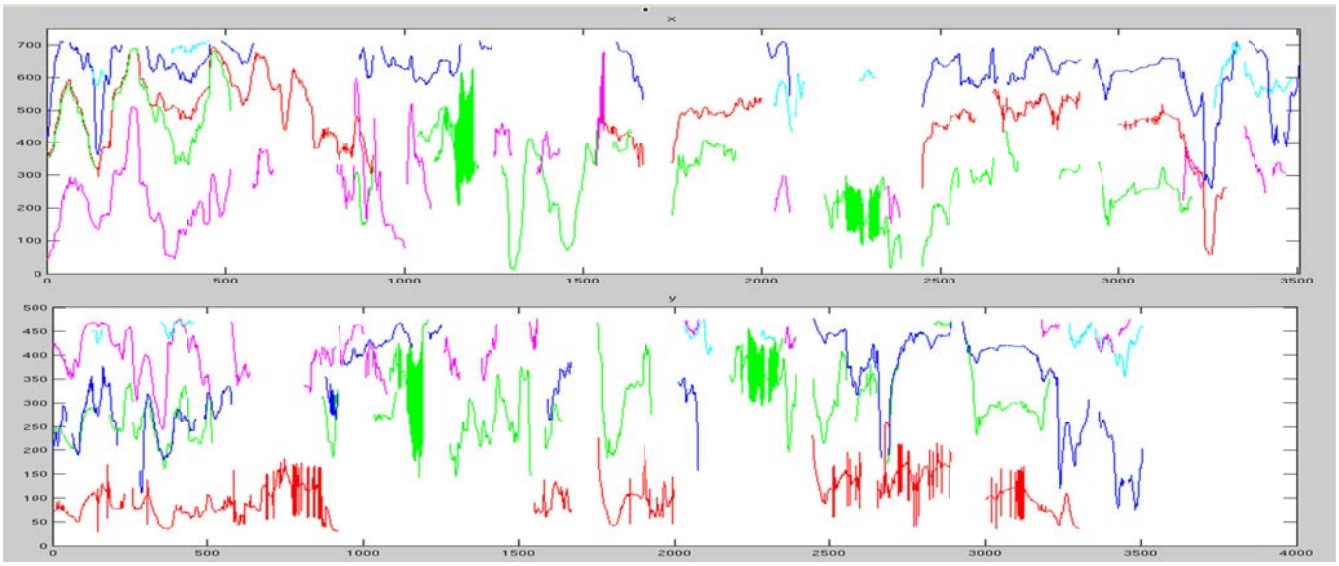

Fig. 10. Blobs center location at X axis and Y axis, top is the X-axis, bottom is Y-axis. Red is face, pink and ligh blue is self left and right hands, green and blue is other person's right and left hands.

Here is the result about tube coordination information. As we can see the red is the head, red line have low Y value, and medial X value. So it means that red tubes was always active in the top center of the view. Other tubes have the same feature.

IV. FUTURE WORK

In the tube generate procedure still need improve; we think HMM algorithm could be a good choice, we can give some pre condition information about the hands movement and face movement. That mean give the HMM some revision about hand and face movement region. That mean self right hand is impossible located at the left side of the self left hand.

For tube classification could use Markov graph to predict the each blobs deification. The accuracy is not very good, still have large improve space.

For optical flow information we can generate the global move value, and extract the local move value. That mean remove camera's movement effect, only left the view object movement.

[1] Ogaki, Keisuke, et al. "Coupling eye-motion and ego-motion features for first-person activity recognition." *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*. IEEE, 2012.

[2] Pirsiavash, Hamed, and Deva Ramanan. "Detecting activities of daily living in first-person camera views." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.

[3] Lee, Yong Jae, Joydeep Ghosh, and Kristen Grauman. "Discovering important people and objects for egocentric video summarization." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.

[4] Fathi, Alireza, Yin Li, and James M. Rehg. "Learning to recognize daily actions using gaze." *Computer Vision–ECCV 2012*. Springer Berlin Heidelberg, 2012. 314-327.

[5] Fathi, Alireza, Jessica K. Hodgins, and James M. Rehg. "Social interactions: A first-person perspective." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.

[6] Kitani, Kris M., et al. "Fast unsupervised ego-action learning for first-person sports videos." *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011.

[7] Starner, Thad, Joshua Weaver, and Alex Pentland. "Real-time american sign language recognition using desk and wearable computer based video." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 20.12 (1998): 1371-1375.

[8] Boykov, Yuri, and Vladimir Kolmogorov. "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.9 (2004): 1124-1137.

[9] Cormen, Thomas H., et al. *Introduction to algorithms*. MIT press, 2001.

[10] Boykov, Yuri Y., and M-P. Jolly. "Interactive graph cuts for optimal boundary & region segmentation of objects in ND images." *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*. Vol. 1. IEEE, 2001.

[11] Li, Yin, et al. "Lazy snapping." *ACM Transactions on Graphics (ToG)* 23.3 (2004): 303-308.

[12] Liu, Ce, et al. "Human-assisted motion annotation." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008.

[13] http://people.csail.mit.edu/celiu/OpticalFlow/