USER-DEFINED BLOB DETECTION AND TRACKING IN CCV

INTRODUCTION

Now a days multitouch is not limited to multiple fingers only. It has become a part of Human Computer Interaction and daily life. People have started using of recognizing objects by attaching the fiducial tag with it. But this is not compatible with all the setups. It can only be possible with DSI and Rear DI setups. Community Core Vision (CCV) has been proved as a boon to the multi-touch computing community because of its efficiency and performance of tracking of fingers and fiducials. In my application I propose to introduce Object/Pattern tracking into CCV in which the object/pattern to be tracked can be completely defined by the user.

The deliverables of this project will be:

i. A tool for selection of templates, where user can select the particular template to match
ii. Template matching integrated into CCV.
iii. Making an MT application to get and send data over Bluetooth/ Wi-Fi network.
iv. Application for demonstration. Where you can transfer images from your MT screen to device just by moving it into the device region on the screen. This will be one use out of many, but will be the best for demonstration purpose.

DETAILED EXPLANATION

It has been seen that any object (e.g. your mobile) when put over any multitouch surface, it creates a particular pattern irrespective of whether the setup is FTIR or LLP. Also I have seen that this pattern remains almost the same irrespective of the position of the object over the screen. Therefore one way of detection/tracking of these objects can be done just by detecting these patterns and then tracking it. Once detection of these objects is done, the objects can be assigned particular IDs for tracking as per user specifies it. As the fiducials have reserved IDs from 0-169 (Reactivision fiducial markers which are recently integrated into CCV), there can be reserved ID space for different objects e.g. say from 170-199 is reserved for mobiles, 200-230 is for special shaped objects or similar. Fingers can be assigned IDs after 300. This will help client program to receive the TUIO data and analyze it as per necessity.
Regarding this project I have done some experiments with my LLP setup. I have taken the 640x480 image of the screen with my mobile phone on it. The results are as follows. Please keep in mind that these images are with 3 lasers and my PS3 eye camera does not have an m12 lens. So it is not very clear and not very focused. Here are some pictures.

[Images of laser patterns with finger and without finger]
Also regarding the detection method I have done some experiments with different detection techniques like SURF (Speed up Robust Features), brute force template matching (with varying angle), and moment matching of the contours. Till now the best result is found with the moment matching with contour detection. This method is found simple and fast. Also the angle detection is possible with this method. The experiment result images are attached in the “Screenshots and UI-Mock-ups” section

**ROADMAP**

The roadmap of the development process as I have arranged it is as follows. This is a rough draft. This may change according to the hurdles which come on the way of programming.

- Create an application over openFrameworks which allows the user to select the template which is to be matched.
  - i. Mouse click and drag on the image to select the template.
  - ii. Generate and save template images in /bin/data/templates folder. Assign IDs according to the user specification.
  - iii. Generate an XML file which lists all the template information like the template ID and weather it’s a mobile or normal object, network address of the ID.
  - iv. Listing of already created templates.
  - v. Deleting any of the previously made templates.
  - vi. Supporting of all the cameras. User specifies resolution.

- Integrate detection and tracking of the created templates into CCV
  - i. For detection purpose the contour matching technique is to be used. Contour match to find the position of the object. Draw rectangle to find the angle of the object.
  - iii. The KNN tracker algorithm (already being used in CCV) is to be used for tracking. As per theory, this seems to be working for such tracking. But in case it is found inefficient, then tracking algorithm to be changed to kalman filter based tracking.
  - iv. The finger blob tracking IDs to be changed so that they start from 300.
  - v. Changing in the TUIO.cpp and TUIO.h to coordinate with tracking and message sending.
  - vi. User can choose what he wants to decide between fingers, fiducials and Patterns. He can select multiple options too. This helps in not wasting processing power.
• Create an application for the detection of the mobile.
  i. Using AS 3.0 or PyMT.
  ii. It will be able to detect and track the objects position and angle.
  iii. Shows up information about the object, must be provided in the template configuration file.
  iv. Images transferred to mobile when an image is dropped in its region.
• A computer application which grabs data from the phone over the Bluetooth/Wi-Fi.
  i. Getting the mobile that is detected from all the Bluetooth/Wi-Fi enabled mobile around. This has to be specified during the template generation.
  ii. Connecting and getting access with authentication.
  iii. Grabbing of the images and transfer of them via Bluetooth/Wi-Fi and saving in a folder. I am planning to do this on python (with LightBlue).

**TIMELINE**

**MAY 24th - JUNE 6th**

• Discussion with the mentors about the roadmap of the project.
• Roadmap to be fixed. Warm up with C++, OpenFrameworks and openCV.
• Finding out the best method for the Template matching invariant of rotation. (for now I am thinking it to be contour matching, but have to try out SURF).
• Finalizing ID structure of the objects.

**JUNE 7th - JUNE 21st**

• Coding starts for the template generating tool in openFrameworks.
• Getting image from the camera at user-specified “width x height”.
• Showing up the image and allowing the user to draw a rectangle over the template, based on which the template is being selected.
• Generation of template data. Storing of either the template image /the contour sequence data in file system. This is to make the loading of data easier by CCV.

**JUNE 22nd - JULY 12th**

• Input of the templates data into CCV.
• Contour matching technique integration into CCV. Trying out of SURF technique.
• KNN Tracking integration. Integration of kalman filter based tracking (if required).
• Dynamic ID assignment in tracking class (instead of auto ID assignment).
Mid-term Evaluation

JULY 16th - JULY 22nd

- Getting the x, y points based on contour.
- TUIO message sending.
- The ID assignment to be taken care of while programming.
- Adding of rotation variable into TUIO data.

JULY 23rd - JULY 29th

- Learning and coding for Bluetooth/Wi-Fi data transfer in python/Java.
- Application design which shows object presence on the screen and tracks it.

JULY 30th - AUGUST 9th

- Coding on the application side for the visual data transfer
- Mostly to be done in AS3 or PyMT
- Integration of ID differentiation into touchlib AS3.0 and PyMT

AUGUST 9th - AUGUST 16th

- Code optimization.
- Documentation.
- Writing usage instruction for end users.
- Setting up future works.

Official Pencil Down- August 16th

Final Evaluation- August 20th

I have plan of moving to another place at the beginning of July. So it may take up 3-4 days in the first week of July for me to settle up in a new place. Except this I don’t have any major plans.
PERSONAL INFORMATION

Name: Amit Kumar Sarangi.

Age: 21

E-Mail: amit.getinfo@gmail.com

Blog URL: http://amitsarangi.wordpress.com

Forum Nick: Amit

Irc Nick: Lunixed

Location: BITS Pilani, Goa Campus, India

Time zone: GMT + 5.30

Education: Pursuing B.E degree in Birla Institute of Technology and Science, Goa Campus in Electronics and Electricals.

Languages: C++, Java, Python, AS 3.0, PHP

I have joined NUIGroup since August, 2009 and since then I have been contributing on different areas, be it application design or helping people on building hardware and software.

I have been involved with open source since I started programming as that is the reason why I got interest in programming. And after I got involved with NUIGroup I have also started contributing to open source projects. I have made 2 open source multitouch Applications, MTRuler (http://amitsarangi.wordpress.com/tag/mtruler/) and MT Editor (http://amitsarangi.wordpress.com/tag/mt-editor/) in AS 3.0. Also I have worked on the Integration of Fiducial detection and tracking into CCV source (http://amitsarangi.wordpress.com/2010/01/29/fiducial-support-in-community-core-vision/). Therefore I am well versed with the CCV codebase.

Also as I have been a part of developer’s community I am well acquainted with the versioning tools (SVN). I am familiar with the NUI community. Also I am the core member of the Center of Software Development (CSD) at BITS Pilani Goa Campus.
I have a working LLP based multitouch table with me. I made this as a part of my ongoing academic project and now I am continuing to research on this. I have presented my project in Open Showcase in Quark 2010, the national technical festival of my college where everyone showcases their project. My project was awarded “The Best Project” among all the projects in this event. Also, my team has got first prize in a robotics event Quark 2008 and 2nd in Quark 2009 (the national technical festival of BITS Pilani Goa Campus). I also have won some awards in paper presentations.

Regarding other programming projects I have done some of the following projects for the campus.

i. Centralized authentication using LDAP server
ii. Campus wide mail server for all the students
iii. Network alarm server client application, a java based network alarm.
iv. PHP modules for a Content Management System

REFERENCE

- Scale Invariant Feature Transform (SIFT) - Computer vision technique for the rotation and scale invariant detection (http://en.wikipedia.org/wiki/Scale-invariant_feature_transform)
- Speeded Up Robust Features (SURF): openSURF library (http://code.google.com/p/opensurf1/)
- LK tracking algorithm (http://vision.eng.shu.ac.uk/mmvlwiki/index.php/Lucas-Kanade_tracker)
- The LightBlue project (http://lightblue.sourceforge.net/) - Python Bluetooth library
- Community Core Vision (http://ccv.nuigroup.com/)
- A Forum Thread on similar concept discussion (http://nuigroup.com/forums/viewthread/2606/)
EXPERIMENT SCREENSHOTS AND MOCK-UPS

Contour detection and Finding of rotation of object

Detection of specific contour (in the “T” window) in an image having different finger-blob type patterns
Specific pattern detection from an image consisting of different type of pattern

Tracking of the patterns (Mock up)
UI for the end users (Mock-up)

CCV Interface with different “Tracking” section (Mock-up)
The interface of “Template Selection Tool” (Mock-up)