Abstract— Due to the ongoing development of low cost computational power it is now plausible to execute image processing techniques on modern workstations. New advances in video processing algorithms provide methods for computer vision functions to be applied in general applications. One useful function is object tracking. The use of visual tracking to follow hand gestures for a user interface is plausible with the use of a common graphics processing unit (GPU). This work in progress explores the components and factors required to implement a visual target tracking system.

I. INTRODUCTION

Computer vision encompasses the general applications and processing of an image input to be used in a computer system. Today cell phones and laptops are being made with embedded cameras making cameras widely available. An increase in camera availability and the ever increasing processor power make computer vision possible for general computer users. An essential function included in computer vision is object tracking. Object or target tracking involves the processing of measurements to form and maintain tracks of an object’s state. An example that would exploit object tracking is the motion of hand movement for a user interface (UI) or a subsystem for sign language interpretation.

The use of visual tracking of the hand for a user interface will be used as a guideline for the design and research. As an example consider entering the letter ‘C’ using a camera. A user could form a pattern using a stylus-based technique [1] as depicted in fig 1. The tracking system would have to extract the hand from each image and determine it position. Some difficulties are revealed in the example such as feature extraction from a cluttered image and a motion that is difficult to predict. The system to be designed would have to overcome the difficulties presented.

In the following sections pertinent issues and solutions regarding visual tracking will be discussed. Section III presents the aims of the research which is followed by a conclusion of the paper.

II. VIDEO TARGET TRACKING

Target tracking is commonly applied to the motion of a target. Using a hand as an example the motion is difficult to model as human motion is often inconsistent. This leads to multimodal tracking which is non-Gaussian and can be non-linear. To achieve motion tracking the shape of the object must be tracked. Considering the tracking of a hand, the shape would not be constant such as an open or closed fist. The changing of shape presents a problem since a simple static model would be insufficient. The system should track an object over a cluttered background. To extract an object from an image features such as edges or corners are found and related to the object. A cluttered background would present features that might look like the object of interest and confuse the system. Issues to be considered are then:

- Tracking of non-linear movement;
- Tracking though a cluttered background; and
- Tracking of a changing shape.

The tracking of position and shape has been dealt with extensively using well known Kalman filter techniques. These methods are often inadequate for tracking on a cluttered background. An effective method for tracking in clutter is the condensation algorithm presented in [2]. The algorithm is similar to particle filter which is discussed further in section II-A below. Klein and Murry [3] express limitations of the condensation algorithm and present the particle filter as an improvement. Image processing and particle filtering are computationally expensive placing a strain on real-time implementation. The GPU offers a solution due to its parallel
nature which is designed for processing images and can be applied to particle filtering.

A. Particle Filter

Particle filters are derived from Bayesian estimation theory using Monte Carlo methods. Bayesian filtering is a general probabilistic estimator that aims at determining the posterior probability density distribution, $p(x_t|y_{1:t-1})$ for a Markov process [4]. The distribution essentially holds information of the state prediction during the tracking process (e.g. the state such as position can be derived from the mean). The filter operates recursively using the transition equation $p(x_t|x_{t-1})$. The Kalman filter is the optimal solution only for linear Gaussian systems since the estimate distribution is modelled as a Gaussian distribution. The particle filter uses discrete samples to approximate the estimate and is not bound by linear models to operate effectively.

Figure 2 shows the basic procedures taken every time step for a particle filter. The current state is estimated by samples with relative weights which represent a distribution shown as $p(x_{k-1}|z_{k-1})$. The samples or particles are propagated using the transition function. Particles are selected so that the weights are equal but still represent the correct distribution. The measurements are then used to update the filter by relating the measurement to the state. The new weighted samples represent the posterior distribution of the predicted state which can be non-Gaussian.

![Fig. 2. Time step for a particle filter using importance sampling.[5]](image)

B. GPU Programming

Graphics processing units were originally produced for limited functionality and high throughput aimed at accelerating graphical procedures. The development of programmable vertex and fragment shaders allowed for general programming on the GPU. The GPU operates by using one shader and applying it to multiple vertices or pixels simultaneously. The GPU is undoubtedly beneficial for image processing. The use of the GPU for particle filter applications is also beneficial since operations can be applied to each sample concurrently. GPUs are not general processing platforms and as such are required to transfer data to the CPU for certain instructions. The transfer creates a bottleneck to the parallel implementation and should be avoided as shown in [6]. Another drawback is the lack of floating point accuracy. Most CPUs support 64-bit precision but GPUs generally support 32-bit or less which can lead to irregularities when porting between the two processor types.

III. Objectives

The general objective of the research is to determine if a viable visual tracking system can be implemented on a modern workstation. The system shall consider tracking in cluttered images without the use of markers. The main focus will be on the estimation techniques involved. Although the Kalman filter does not cater for non-Gaussian estimations it does require less computational effort. Given that the system can be tracked using a Kalman filter it would outperform the Particle filter in speed. A comparison of Kalman filter and Particle filters will be conducted relative to the system. The comparison shall include aspects such as speed and robustness.

An aspect that will also be considered is the use of the GPU on the workstation. The system cannot dedicate all the resources to the tracking system for applications that use it as a subsystem such as a user interface. An investigation shall be made into the division of resources between tracking and general use.

IV. Conclusion

Computer vision has been greatly improving over the past years mainly in the robotics field. This work in progress explains the basis of a visual object tracking system applicable for common workstations. The research is in its infancy but shows that a system could be developed for home use. Following research will show whether near real-time, robust tracking can be accomplished on a workstation and allow for concurrent user applications.

REFERENCES


Mark Evans received his B.Eng. in Electrical and Electronic Engineering with endorsement B.Sc. in Information Technology from the University of Johannesburg in 2008. He is currently working towards an M.Eng. in Electrical and Electronic Engineering.