Proceedings of the 5th WSEAS Int. Conf. on SIGNAL, SPEECH and IMAGE PROCESSING, Corfu, Greece, August 17-19, 2005 (pp227-230)

Trainable Gesture Recognition Framework for Virtual Keyboard

HAFIZ ADNAN HABIB, MUID MUFTI
Computer Engineering Department
University of Engineering & Technology
Taxila, Punjab
PAKISTAN

Abstract: this research presents gesture recognition based virtual keyboard functional framework. Fingertip trajectory based key stroke detection, articulation feature vector based key association are basic building blocks for the framework. This framework is trainable and learns the trajectories of fingertips of different users to make key stroke and associated articulation feature vector for pressing particular key by certain finger.

Keywords: virtual keyboard, gesture recognition, framework, trainable system, fingertip trajectory.

Introduction:

Keyboards have mechanical keys which are pressed by fingertips of human beings to type a particular key. Use of such keyboards is limited for data entry in portable devices such as mobile phones and PDA's etc. Variety of specially designed keyboards is developed for such devices [1]. These designs rely upon variety of technologies and sensors: gestured gloves, accelerometers, sensing switches and sensing pads etc. Virtual keyboards are also designed by computer vision techniques. In such designs, a virtual keyboard is projected and video analysis are performed to detect the keys pressed [2], [3]. A keyboard is projected on any surface by holographic diffraction pattern to diffract a single beam of laser light directly into the pattern to be projected. Position of finger striking this projected IR surface is computed by triangulation method. The analysis system infers finger-to-surface contact from motion of fingers. Event classification determines the type of action: landing, hold, move, take-off. Triangulation transforms image points into keyboard positions, a table then maps to the identity of the key associated with that position. Key identity and event type determine the appropriate keyboard event.

Aims & Objectives

This research work presents the theoretical development of virtual keyboard based upon gesture recognition paradigm. Gesture analysis is performed in understanding sign language, activity recognition, augmented reality, virtual interfaces and human computer interaction. Virtual keyboard framework is essentially an effort towards gesture recognition based human computer interaction.

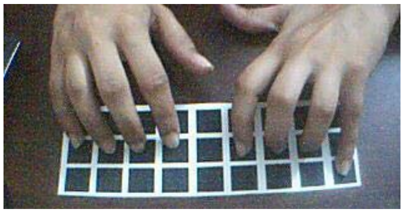
Design Framework

Activities in this intelligent system are divided in following sub-activities: hand pose estimation, finger-tip registration, finger-tip trajectory generation, trajectory parameterization, key-stroke event detection, estimation of pressed key from trajectory, finger articulation feature vector extraction, and key classification. Block diagram of these activities is shown in fig.1.

Posture Mode Selection

Hand pose is computed for initialization and posture of hand is correlated to pre-stored postures of hands. Different users have different styles in which they place their hands to type through keyboard. Also different users have variation in their skin color, size of fingers and size of nails / fingertips. System learns these parameters in supervised learning fashion for each specific user.

Proceedings of the 5th WSEAS Int. Conf. on SIGNAL, SPEECH and IMAGE PROCESSING, Corfu, Greece, August 17-19, 2005 (pp227-230)



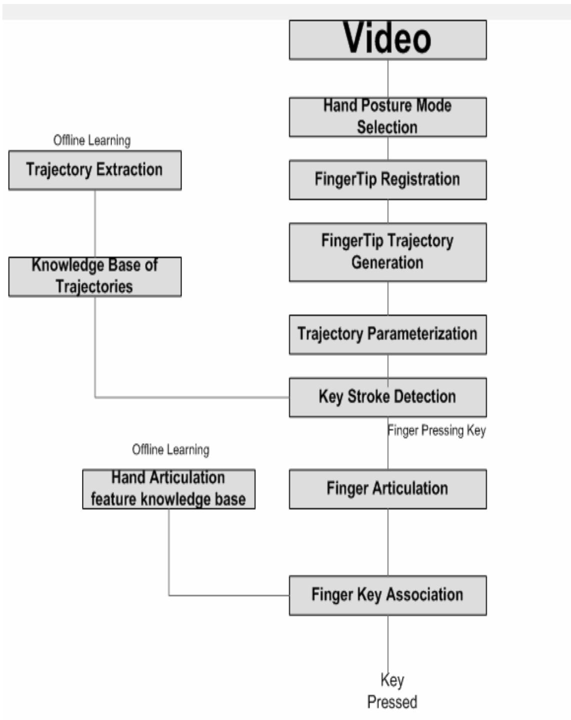


Fig. 1: Block Diagram

Finger-tip Registration

Finger-tips are pointing devices capable to press keys on keyboard. After hand pose estimation, eight finger-tips are registered in sensor domain. Thumbs were excluded in this experimentation for key pressing activity since experimentation was restricted to three rows of alphabets and were press-able by eight keys.

Finger-tip Trajectory Generation

No key can be pressed in static way. Even for repetitive pressing of particular key, it is required that finger-tip should press key by making motion each time. Since the design was to press keys in three rows only, so it also becomes un-necessary to press another key while one key is already pressed, the usual case with shift key which is kept low while another key is also pressed.

Motion is detected at first stage. Only those motions are considered which comprises group of pixels rather isolated pixels to avoid noise. Motion is recorded to produce eight trajectories: one trajectory for each finger-tip.

Finger-tip Trajectory Parameterization

This intelligent system has to detect key strokes from trajectories of each finger-tip. Motion types have to be considered e.g. translation, rotation, cycloid, projectile etc. [5]. Type of motion is important consideration in parameterization of motion. It was observed that finger-tips mostly make curve translations rather than straight line translations. Fingers in each hand make movement for each key. Ideally one finger-tip should move to press desired key, however practically all fingers of hand make some motion. Motion made by key pressing finger-tip is mostly distinct. In analysis of such motions, one has to opt for relative motion or common motion. In this experimentation, it was found that computing relative motion is unnecessary. Computation of motion of each finger is sufficient to achieve ultimate goal of finding key stroke event made by particular finger-tip. Speed and direction are two very obvious features for motion analysis. However there are many other features proposed in literature, but it was experienced that these two features are sufficient to extract key-stroke events. [5]

Key-Stroke Detection

All eight trajectories of finger-tips in parameterized form are delivered to this stage. This is basically a motion boundary problem. Usually there are five type of motion boundaries: smooth starts, smooth stops, pauses, impulse starts and impulse stops. In this virtual keyboard design, it was found that there is to and fro motion, while this to and fro motion is separated by a pause in motion of small interval. For this particular information to be extracted from trajectory, amount of length of trajectory is of important consideration. While initial experimentation was restricted to typing the keys in three rows only, so a range of trajectory length is learned for each specific user to analyze key stroke events in parameterized finger-tip trajectories.

This key-stroke detection reveals information about particular key which is responsible for making such movement. Outcome of this stage is many to one finger-tip responsible for trajectory generation. Here benefit of considering trajectories of individual finger-tips is evident that key-stroke detection also points the finger-tip responsible for key-stroke generation. This simplifies the classifier design to extract particular key as the options are reduced from 26 alphabets to fewer alphabets which are associated to particular finger-tip.

Key Classification:

This part of work is dependent upon two stages: key estimation from finger-tip trajectory and finger articulation information. Keys can be estimated from finger-tip trajectory but it did not prove very productive and it also did not simplify the problem. However finger articulation vector is incorporated. This computes the shape features of each finger-tip and it operates in parallel to above mentioned stages. Particular key generating key-stroke event is revealed by key-stroke detection process. Articulation feature information of this particular key is taken form finger articulation feature vector. Simple distance classifier is incorporated to classify the key generating key-stroke.

Learning Architecture of Frame-work

The proposed frame-work learns following information: hand pose, finger-tip for robust trajectory generation, key-stroke trajectories for all keys related to each finger-tip.

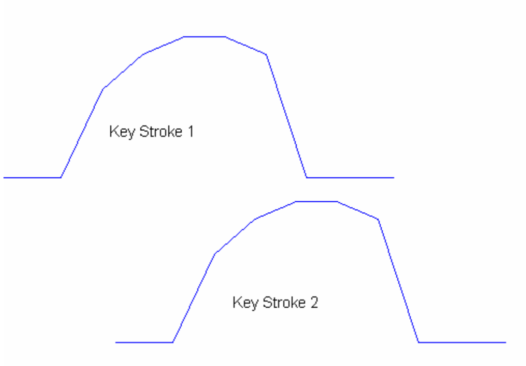
Experimentation Setup

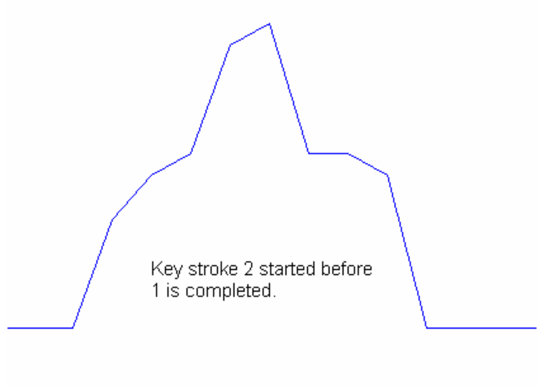
Video sequences are captured in laboratory environment under normal lighting conditions without incorporating any ambient or special purpose light sources. A printed keyboard is placed for user reference. In initial experimentation, only alphabet keys are printed and thus restricting keyboard design to three rows of alphabet keys but placement follows standard arrangement to avoid confusion. This restriction was applied for initial experimentation.

Conclusion & Future Work

The results of this frame work were encouraging. Initial experimentation was restricted to three rows of keys. This experimentation should be conducted over printed versions of standard keyboards with complete functionality. It is more challenging in every aspect to develop key-stroke detection algorithm from many possible trajectories and wider set of possible motions due to increase in number of keys.

At this stage, it was considered that user presses the keys in following steps: hand staying in initialization form, key press by hand motion and attaining back to initialization posture. It was considered that either hand will make movement to press a key and come back to the pose of initialization. At other time, it will make movement, press a key and attain initialization pose. This simplified key stroke detection from trajectories. In future work, it is aimed that typing process will start from initialization phase. Then hands have not to attain initialization posture after each key press, rather key pressing will be done in continuous fashion and analysis of keystrokes will be done accordingly. This will lead to more natural typing possible by gesture recognition.





Reference:

[1] Mathias Kolsch, Matthew Turk "keyboards without keyboards: a survey of virtual keyboards".

[2] Carlo Tomasi, Abbas Raffi, Ilhami Torunoglu "full-size projected keyboard for handheld devices" 2003.

[3] www.vkb.co.il

[4] Helena Roeber, John Bacus, Carlo Tomasi. "Typing in thin air Canesta projection keyboard – a new method of interaction for handheld devices", CHI 2003.

[5] Claudette Cedras, Mubarak Shah "Motion based recognition; a survey".