CHAPTER 4

IMAGE OPERATION’S FOR TRACKING FACE AND VIDEO FRAMES

In this chapter discussed about the Face identification (face recognition) and detection including live and still video streams (For a variety of real time scenarios) with their applications and corresponding tools is discussed.

4.1 INTRODUCTION

A facial recognition system is a computer application for automatically identifying or verifying a person from a digital image or a video frame. This is done by comparing selected facial features from the image and facial details from database. It is widely used in security systems and can be compared to other biometrics such as fingerprint or eye iris recognition systems. Some facial recognition algorithms identify facial features by extracting landmarks, or features, from an image of the subject's face. For example, an algorithm may analyze the relative position, size, and/or shape of the eyes, nose, eye movements, cheekbones and jaw. These features are then used to search for other images with matching features. Some algorithms normalize a gallery of face images and then compress the face data, only saving the data in the image that is useful for face recognition. A probe image is then compared with the face data. One of the earliest successful systems is based on template matching techniques applied to a set of salient facial features, providing a sort of compressed face representation. Recognition algorithms can be divided into two main approaches, geometric, (looks at distinguishing features) or photometric, (statistical approach) that distills an image into values and compares the values with templates to eliminate variances. Widely used recognition algorithms include Principal Component Analysis (PCA) using eigenfaces, Linear Discriminate Analysis (LDA), Elastic Bunch Graph Matching using the Fisher face algorithm, the Hidden Markov model, the Multi linear Subspace Learning using tensor representation, and the neuron motivated dynamic link matching.

4.2 3 DIMENSIONAL RECOGNITION
3-dimensional recognition claimed to achieve improved accuracies in face recognition. This technique uses 3D sensors to capture information about the face. This information is then used to identify distinctive features on the surface of a face, such as the contour of the eye sockets, nose, and chin. One advantage of 3D facial recognition is that it is not affected by changes in lighting like other techniques. It can also identify a face from a range of viewing angles, including a profile view. Three-dimensional data points from a face vastly improve the precision of facial recognition. 3D research is enhanced by the development of sophisticated sensors that do a better job of capturing 3D face imagery. The sensors work by projecting structured light onto the face. More image sensors can be placed and each sensor captures a different part of the spectrum. Even a perfect 3D matching technique could be sensitive to expressions.

4.2.1 SKIN TEXTURE ANALYSIS

Salient Features include

(i) Using the visual details of the skin as captured in scanned images.
(ii) Maps the unique lines, patterns, and spots apparent in a person’s skin into a mathematical space.

4.2.2 TEMPLATE MATCHING

Salient Features include

(i) Used for finding small parts of an image
(ii) Used in manufacturing
(iii) Used to navigate a mobile robot
(iv) Used to detect edges in images

4.2.3 FEATURE BASED APPROACH

Salient Features include

(i) Computationally efficient when working with source images of larger resolution
(ii) Determine the best matching location.

4.2.4 TEMPLATE BASED APPROACH
Salient Features include

(i) Used for when the bulk of the template image constitutes the matching image
(ii) Reduce the number of sampling points by reducing the resolution of the search and template images by the same factor and performing the operation on the resultant downsized images (multi resolution, or pyramid, image processing).

4.2.5 MOTION TRACKING AND OCCLUSION HANDLING

Salient Features include

In instances where the template may not provide a direct match, it may be useful to implement the use of eigenspaces templates that detail the matching object under a number of different conditions, such as varying perspectives, illuminations, color contrasts, or acceptable matching object “poses”. For example, if the user was looking for a face, the eigenspaces may consist of images (templates) of faces in different positions to the camera, in different lighting conditions, or with different expressions.

4.2.6 TEMPLATE BASED MATCHING AND CONVOLUTION

(Linear Spatial Filtering)

Salient Features include

(i) Using a convolution mask
(ii) The convolution output will be highest at places where the image structure matches the mask structure (Filter Mask).

4.3 METHODOLOGY

This method is normally implemented by first picking out a part of the search image to use as a template: We will call the search image $S(x, y)$, where $(x, y)$ represent the coordinates of each pixel in the search image. We will call the template $T(x_t, y_t)$, where $(x_t, y_t)$ represent the coordinates of each pixel in the template. We then simply move the center (or the origin) of the template $T(x_t, y_t)$ over each $(x, y)$ point in the search image and calculate the sum of products between the coefficients in $S(x, y)$ and $T(x_t, y_t)$ over the whole area spanned by the template. As all possible positions of the template with respect to the search image are considered, the position with the highest score is the best position. For
example, one way to handle translation problems on images, using template matching is to compare the intensities of the pixels, using the SAD (Sum of absolute differences) measure. A pixel in the search image with coordinates \((x_s, y_s)\) has intensity \(I_s(x_s, y_s)\) and a pixel in the template with coordinates \((x_t, y_t)\) has intensity \(I_t(x_t, y_t)\). Thus the absolute difference in the pixel intensities is defined as

\[
\text{Diff}(x_s, y_s, x_t, y_t) = | I_s(x_s, y_s) - I_t(x_t, y_t) |.
\]

\[
SAD(x, y) = \sum_{i=0}^{T_{rows}} \sum_{j=0}^{T_{cols}} \text{Diff}(x + i, y + j, i, j) \quad (4.1)
\]

The mathematical representation of the idea about looping through the pixels in the search image as we translate the origin of the template at every pixel and take the SAD measure is the following

\[
\sum_{x=0}^{S_{rows}} \sum_{y=0}^{S_{cols}} SAD(x, y) \quad (4.2)
\]

\(S_{rows}\) and \(S_{cols}\) denote the rows and the columns of the search image and \(T_{rows}\) and \(T_{cols}\) denote the rows and the columns of the template image, respectively. In this method the lowest SAD score gives the estimate for the best position of template within the search image. The method is simple to implement and understand, but it is one of the slowest methods.

### 4.3.1 IMPLEMENTATION

(i) Applied on grey images

(ii) The final position in this implementation gives the top left location for where the template image best matches the search image.

**Pseudo Code**

```plaintext
minSAD = VALUE_MAX;
// loop through the search image
for(int x =0; x <=S_rows-T_rows; x++)
{
    for(int y =0; y <=S_cols-T_cols; y++)
    {
        SAD =0.0;
        // loop through the template image
        for(int j =0; j <T_cols; j++)
```
for(inti=0;i<T_rows;i++)
{
    pixelp_SearchIMG= S[x+i][y+j];
    pixelp_TemplateIMG= T[i][j];
    SAD +=abs(p_SearchIMG.Grey-p_TemplateIMG.Grey);
}
// save the best found position
if(minSAD> SAD )
{
    minSAD= SAD;
    // give me min SAD
    position.bestRow= x;
    position.bestCol= y;
    position.bestSAD= SAD;
}
}

One way to perform template matching on color images is to decompose the pixels into their color components and measure the quality of match between the color template and search image using the sum of the SAD computed for each color separately.

### 4.3.2 SPEEDING UP THE MATCHING PROCESS

Conventionally spatial filtering is constrained to dedicated hardware solutions due to the computational complexity of the operation. This constrained be removed by filtering it in the frequency domain of the image, referred to as 'frequency domain filtering,' this is done through the use of the convolution theorem.

Another way of speeding up the matching process is through the use of an image pyramid. This is a series of images, at different scales, which are formed by repeatedly filtering and sub sampling the original image in order to generate a sequence of reduced resolution images. These lower resolution images can then be searched for the template (with a similarly reduced resolution), in order to yield possible start positions for searching at the larger scales. The larger images can then be searched in a small window around the start position to find the best template location. Other methods can handle problems such as translation, scale, image rotation and even all affine transformations.
Improvements can be made to the matching method by using more than one template (Eigenspaces), while other templates can have different scales and rotations. It is also possible to improve the accuracy of the matching method by hybridizing the feature-based and template-based approaches. Naturally, this requires that the search and template images have features that are apparent enough to support feature matching. Alternative methods which are similar include 'Stereo matching', 'Image registration' and 'Scale-invariant feature transform'.

4.4 APPLICATIONS OF TEMPLATE MATCHING

Thus the figure 4.1 shows the Deformable Template Matching pattern recognition techniques. Template matching has various applications and is used in such fields as face recognition (see facial recognition system) and medical image processing. Systems have been developed and used in the past to count the number of faces that walk across part of a bridge within a certain amount of time. Other systems include automated calcified nodule detection within digital chest X-rays. Recently, this method was implemented in geo statistical simulation which could provide a fast algorithm.

![Figure 4.1 Deformable Template](image)

Elastic matching is one of the pattern recognition techniques in computer science. Elastic matching (EM) is also known as deformable template, flexible matching, or nonlinear template matching. Elastic matching can be defined as an optimization problem of two-dimensional warping specifying corresponding pixels between subjected images.
In mathematics and computer science, an optimization problem is the problem of finding the best solution from all feasible solutions. Optimization problems can be divided into two categories depending on whether the variables are continuous or discrete. An optimization problem with discrete variables is known as a combinatorial optimization problem. In a combinatorial optimization problem, we are looking for an object such as an integer, permutation or graph from a finite (or possibly countable infinite) set.

4.5. GESTURES RECOGNITION

Recent research studies have shown the great contribution that serious games offer in the area of biomedical rehabilitation of injured persons. Following this trend, we intend to develop a friendly software application for monitoring the physical therapy movements of patients suffering from severe motor disabilities. The correctness of the rehabilitation exercises is verified by applying image processing techniques.

4.5.1 INPUT DEVICES FOR GESTURES RECOGNITION

The ability to track a person's movements and determine what gestures they may be performing can be achieved through various tools. Although there is a large amount of research done in image/video based gesture recognition, there is some variation within the tools and environments used between implementations.

- **Wired gloves.** These can provide input to the computer about the position and rotation of the hands using magnetic or inertial tracking devices. Furthermore, some gloves can detect finger bending with a high degree of accuracy (5-10 degrees), or even provide haptic feedback to the user, which is a simulation of the sense of touch. The first commercially available hand-tracking glove-type device was the Data Glove, a glove-type device which could detect hand position, movement and finger bending. This uses fiber optic cables running down the back of the hand. Light pulses are created and when the fingers are bent, light leaks through small cracks and the loss is registered, giving an approximation of the hand pose.

- **Depth-aware cameras.** Using specialized cameras such as structured light or time-of-flight cameras, one can generate a depth map of what is being seen through the camera at a short range, and use this data to approximate a 3D representation of what is being seen. These can be effective for detection of hand gestures due to their short range capabilities.
• **Stereo cameras.** Using two cameras whose relations to one another are known, a 3d representation can be approximated by the output of the cameras. To get the cameras’ relations, one can use a positioning reference such as a lexian-stripe or infrared emitters. In combination with direct motion measurement (6D-Vision) gestures can directly be detected.

• **Controller-based gestures.** These controllers act as an extension of the body so that when gestures are performed, some of their motion can be conveniently captured by software. Mouse gestures are one such example, where the motion of the mouse is correlated to a symbol being drawn by a person's hand, as is the Wii Remote or the Myo, which can study changes in acceleration over time to represent gestures. Devices such as the LG Electronics Magic Wand, the Loop and the Scoop use Hillcrest Labs' Free space technology, which uses MEMS accelerometers, gyroscopes and other sensors to translate gestures into cursor movement. The software also compensates for human tremor and inadvertent movement. AudioCubes are another example. The sensors of these smart light emitting cubes can be used to sense hands and fingers as well as other objects nearby, and can be used to process data. Most applications are in music and sound synthesis,[30] but can be applied to other fields.

• **Single camera.** A standard 2D camera can be used for gesture recognition where the resources/environment would not be convenient for other forms of image-based recognition. Earlier it was thought that single camera may not be as effective as stereo or depth aware cameras, but some companies are challenging this theory. Software-based gesture recognition technology using a standard 2D camera that can detect robust hand gestures, hand signs, as well as track hands or fingertip at high accuracy has already been embedded in Lenovo’s Yoga ultrabooks, Pantech’s Vega LTE smartphones, Hisense’s Smart TV models, among other devices.

### 4.6 VARIOUS ALGORITHMS USED FOR GESTURES RECOGNITION
Figure 4.2 Classification of Gestures Recognition

The figure 4.2 shows the different ways of tracking and analyzing gestures exist, and some basic layout is given is in the diagram above. For example, volumetric models convey the necessary information required for an elaborate analysis, however they prove to be very intensive in terms of computational power and require further technological developments in order to be implemented for real-time analysis. On the other hand, appearance-based models are easier to process but usually lack the generality required for Human-Computer Interaction.

Depending on the type of the input data, the approach for interpreting a gesture could be done in different ways. However, most of the techniques rely on key pointers represented in a 3D coordinate system. Based on the relative motion of these, the gesture can be detected with a high accuracy, depending of the quality of the input and the algorithm’s approach.

In order to interpret movements of the body, one has to classify them according to common properties and the message the movements may express. For example, in sign language each gesture represents a word or phrase. The taxonomy that seems very appropriate for Human-Computer Interaction has been proposed by Quek in "Toward a Vision-Based Hand Gesture Interface". He presents several interactive gesture systems in order to capture the whole space of the gestures: 1. Manipulative 2. Semaphoric 3.
Some literature differentiates 2 different approaches in gesture recognition: a 3D model based and an appearance-based. The foremost method makes use of 3D information of key elements of the body parts in order to obtain several important parameters, like palm position or joint angles. On the other hand, Appearance-based systems use images or videos for direct interpretation. A real hand (left) is interpreted as a collection of vertices and lines in the 3D mesh version (right), and the software uses their relative position and interaction in order to infer the gesture.

### 4.7 3D MODEL BASED ALGORITHMS

The 3D model approach can use volumetric or skeletal models, or even a combination of the two. Volumetric approaches have been heavily used in computer animation industry and for computer vision purposes. The models are generally created of complicated 3D surfaces, like NURBS or polygon meshes. The drawback of this method is that is very computational intensive and systems for live analysis is still to be developed. For the moment, a more interesting approach would be to map simple primitive objects to the person’s most important body parts (for example cylinders for the arms and neck, sphere for the head) and analyse the way these interact with each other. Furthermore, some abstract structures like super-quadrics and generalised cylinders may be even more suitable for approximating the body parts. The exciting thing about this approach is that the parameters for these objects are quite simple. In order to better model the relation between these, we make use of constraints and hierarchies between our objects. The skeletal version (right) is effectively modelling the hand (left). This has fewer parameters than the volumetric version and it's easier to compute, making it suitable for real-time gesture analysis systems.

### 4.8 SKELETAL BASED ALGORITHMS

Instead of using intensive processing of the 3D models and dealing with a lot of parameters, one can just use a simplified version of joint angle parameters along with segment lengths. This is known as a skeletal representation of the body, where a virtual skeleton of the person is computed and parts of the body are mapped to certain segments. The analysis here is done using the position and orientation of these segments and the relation between each one of them (for example the angle between the joints and the relative position or orientation)

Advantages of using skeletal models:

- Algorithms are faster because only key parameters are analyzed.
- Pattern matching against a template database is possible
• Using key points allows the detection program to focus on the significant parts of the body.

These binary silhouette (left) or contour (right) images represent typical input for appearance-based algorithms. They are compared with different hand templates and if they match, the correspondent gesture is inferred.

4.9 APPEARANCE BASED MODELS

These models don’t use a spatial representation of the body anymore, because they derive the parameters directly from the images or videos using a template database. Some are based on the deformable 2D templates of the human parts of the body, particularly hands. Deformable templates are sets of points on the outline of an object, used as interpolation nodes for the object’s outline approximation. One of the simplest interpolation functions is linear, which performs an average shape from point sets, point variability parameters and external deformations. These template-based models are mostly used for hand-tracking, but could also be of use for simple gesture classification.

A second approach in gesture detecting using appearance-based models uses image sequences as gesture templates. Parameters for this method are either the images themselves, or certain features derived from these. Most of the time, only one (mono scopic) or two (stereoscopic) views are used.