2006

Hand Posture Recognition in Sign Language Using Shape Distributions

Eamonn Young
Technological University Dublin, eamonn@webeireann.com

Gary Clynch
Technological University Dublin, gary.clynch@tudublin.ie

Hand Posture Recognition in Sign Language Using Shape Distributions

Eamonn Young and Gary Clynch
Department of Computing,
ITT Dublin, Tallaght, Dublin, Ireland
E-mail: eamonn@webeireann.com, gary.clynch@it-tallaght.ie

Abstract

A shape distribution is a histogram used to uniquely identify different shapes. The histogram is produced by taking random distances on the surface of a shape or object. Theoretically, each shape or object should produce a unique histogram, as the distribution of distances for each shape should be different. Shape distributions have recently been implemented in a number of object recognition areas. They are an attractive method as they are inherently simple, fast and generic. This paper presents the results of research undertaken on the application of shape distributions for the purpose of sign language recognition. There are four main elements that need to be undertaken in building a fully operational sign language recognition system, namely: posture, position, orientation and motion. It is the first of these components that this paper addresses.

1 Introduction.

Sign language is a means of communication for deaf people all over the world. It employs the use of different hand gestures to correspond with words in the spoken-language. Also, certain postures relate to the alphabet, so that if a certain word does not exist in the sign language, a person's name, for example, it can be spelt out using this signed alphabet. Similar to a spoken language, the sign language differs from one country to the next. It is the intention of this research to focus on the alphabet of the English Sign Language.

The area of sign language recognition has attracted attention because of its numerous application potentials [3]. A sign language recognition system would consist of an input method to acquire the sign language being performed, for example a video camera. This input could then be processed and analysed by a computer. Several possibilities would be available at this point. One potential application area for a sign language recognition system would be for automatic translation from sign language into spoken or written words. Another area where such a system might be useful would be in human-computer interaction, for example: performing sign language as an input method rather than using a keyboard. Further application areas include industrial environments, whereby levers might be replaced by the more intuitive use of sign language; or a sign language recognition system could also be used in robot communication.

The remainder of this paper is divided into four sections. Section 2 discusses a number of existing techniques that have been used for the sign language recognition. Each system is quite recent and different in its approach. Section 3 introduces the concept of shape distributions and how they have been implemented as a hand posture recognition technique. A detailed discussion is provided on the underlying concept of the technique as well as a description of how it works. Subsequently the tests and experiments that have been conducted on the system are presented. Finally, conclusions are provided as well as some suggestions for future work that might be conducted to further this particular line of research.

2 Related Work.

In order to build a fully capable sign language recognition system, there are four elements which need to be recognised: motion, orientation, position, and posture of the hand [8]. A system has been developed to recognise the motion a hand produces in

three-dimensional space while performing sign language recognition using Finite State Machines [14]. The system is performed in real-time, as FSMs lend themselves to real-time applications, allowing for fast comparison checks as the user performs the gestures. This system is an extension of a previous one developed in a two-dimensional capacity [7]. The three-dimensional system presents some advantages over it's two-dimensional predecessor. The main drawback of the two-dimensional system was that it had no perception of depth as it employs monocular vision. Three-dimensional image capture easily lends itself to depth perception as well as overcoming the obstacle of occlusion by employing binocular vision - like human vision. An initialisation stage takes input and processes it to cater for users of different height and at different distances from the camera, thus providing tolerance to scale differences. A colour information technique is then implemented to segment the head and hand from the image [13]. The head is then used as a point of origin, while the hand can be traced as it moves throughout the gesture. The 3D space is divided up into regions. A gesture is performed repeatedly and information is stored about each gesture. After the gesture is performed a number of times, clustered information provides an average gesture. The stored information is then used as a comparison operand for recognising an input gesture. The information stored in each FSM includes a deviation parameter and temporal information. The former parameter allows for the gesture to be performed without having to pass through the *exact* path that has been recorded. The latter parameter records a maximum and minimum time the hand remains in a certain position for each gesture.

Further motion techniques include a method for dealing with self-occlusion [11]. This might occur if the hand were to rotate in such a way that neither camera could distinguish a finger that might be obstructed from vision by another finger or palm of the hand. A tracking algorithm is used to recognise when a finger obstructs another. It can then effectively remember where the other finger was.

Another element of sign language recognition is recognising a posture that a hand is in. Due to the large amount of positions and orientations a human hand can perform, hand posture identification is quite a complicated task [15]. Neural networks have been applied as a posture recognition technique achieving 90.45% successful matching [16]. It worked by calculating the slope of the hand, the angles of the fingers and the distance of fingers from the centre of the hand. The system was tested with 158 test hand images 1580 times and was trained to recognise 26 postures.

One early technique used for hand posture recognition utilised three-dimensional models of a human hand [4]. The hand model is based on a design proposed to deal with self-occlusions in articulated objects [11]. The model hand is constructed from truncated cones, which represent fingers; and circles for joints. This allows the model of the hand to be manipulated just like a human hand, although extra constraints might need to be added to prevent the hand being able to adopt a position impossible for a human hand. Thus, the shape of the model could be adjusted to match that of an input image of a human hand. This is accomplished by implementing ICP (Iterative Closest Point) [18]. The theory is tested on a simulation and produces an accurate match. However, while the model appears to converge to the correct position in a real-case scenario, it is difficult to measure the exact accuracy.

Another method for hand posture recognition investigated a technique of silhouette-matching [17]. This involves taking a silhouette image of a posture and using it for comparison, rather than using a hand. A drawback of this technique is orientation: if the hand performs the same gesture, but rotates slightly, it creates a different silhouette. This problem was reasonably alleviated in later research [5]. It was proposed to have 72 silhouettes taken of each hand posture - one for every 5 degrees rotation around the x-axis. However, this would also have to be done for the remaining two axes, resulting in a large, 72^3, number of comparisons per posture.

Considerable improvements were produced by combining the two aforementioned approaches into a hybrid method [15]. The system used 36 hand models based on the 36 postures used in Irish Sign Language. These hand models could be orientated and rotated by the system in real-time. A silhouette for each gesture is captured by employing the three-dimensional image capture discussed earlier [14]. By merging these two techniques, it greatly decreases the number of templates required for comparison. Comparison was performed by implementing the Chamfer distance algorithm [1]. As the system worked in a three-dimensional capacity, two cameras were used for recognition. From a total of four input images, the system correctly identified one image with both cameras; two images with only one camera; and failed to recognise another image with both cameras; thus producing a recognition rate of 50%.

3 The Technique.

The concept of shape recognition by implementing shape distributions was first introduced as a three-dimensional object recognition technique [10]. The algorithm works by choosing two random points on the surface of an object and calculating the distance between these two points. This distance is then recorded in a corresponding bin of a histogram. Each time this distance occurs, the value of the bin is incremented. This applies for all of the random distances that might occur on the object. This operation is repeated numerous times until a signature is produced for that object. Theoretically, every different shape will have a different distribution of distances, thus producing a unique signature or histogram for each object. Let us consider a basic example:

Imagine we wish to produce a histogram for a straight line - as an analogy, it might help to visualise a ruler. If a ruler is 30cm long, it means that the distance of 30cm occurs on the ruler only once. The distance of 1cm occurs on the ruler 30 times. If we were to choose a random distance on the ruler, it is more probable that we would select a distance of 1cm than 30cm as 1cm occurs more often, in other words, there is a greater chance of us choosing a distance of 1cm. This means that we would produce a histogram similar to that in Figure 1. This histogram is analogous to all straight lines.

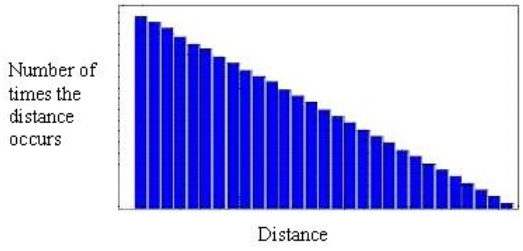

Figure 1: Histogram of a straight line

The researchers found that there was a linear trade-off between accuracy and time. In other words, the more random calculations performed, the higher the rate of accuracy. The decision was made to err on the side of robustness and take a large number of samples or calculations, 1024^2, with 1024 bins. 133 3D objects were downloaded from various 3D databases available on the Internet. The objects consisted of regular, everyday three-dimensional objects, for example: belts, cars, mugs etc. Five separate, but similar algorithms of low complexity were implemented. Each algorithm involves a simple calculation of either distance, area or volume. The *D2* algorithm producing the best results, i.e. a 66% accuracy rating. In other words, the system correctly identified an input object to it's counterpart in the database of objects two times out of three. The primary downfall of the system was in distinguishing between objects of similar shape, for example, a missile and a submarine. Such objects produced histograms that were too similar in shape to be distinguishable at the graph comparison stage. However, it was proposed that the system could be implemented as a pre-classification stage - narrowing the extent of a search to a number of objects of similar shape. These objects could then be further examined using a detailed object recognition technique. This would save a lot of time compared to a situation where one were to use the latter detailed technique from the start.

It was suggested that the algorithm might be useful on other application areas. Since then, it was used - quite effectively - in a protein-structure recognition capacity [2]. The algorithm successfully identified 8 protein structures of the same family from 26,600 domains, exhibiting a classification accuracy of 98% for CATH homologous families. The technique has also been incorporated into a shape estimation system with significant results [9].

The research presented in this paper investigated whether the aforementioned shape recognition technique might prove beneficial as a component of a larger sign language recognition system. There are 26 letters in the alphabet, and consequently 26 hand postures for the sign language alphabet. Each posture involves the hand generating a different shape. If each posture is unique enough, it follows that each signature will be unique and thus provide us with an accurate posture recognition system.

The system can be broken into two stages. Initially the images of a hand are taken. These images consist of a hand and a background. The hand is segmented from the background by identifying pixels that are skin colour. These images are then processed so that a histogram is produced. These histograms provide a template for comparing against. This process could be done off-line as preparation for the actual recognition stage. The second stage involves testing some input images. At this stage, an image of a hand is input to the system. It is compared against all templates stored in the system and a match is returned.

In the early stages of the research, 26 images were

taken of a hand performing the 26 postures of the sign language alphabet. These images are used as a sample set for the system and also the test set. The images are stored locally as JPEGs on the computer that is running the system. Each image is then rasterized and preprocessed to eliminate any "noise" that may be present. The height and width of each image is noted and stored in order to scale the distances that will be calculated for each image. The image is then processed to locate all of the points or pixels on the hand. It is then cropped for performance purposes so that the edges of the hand all lie on the borders of the image. This step reduces the area of the image that does not contain the hand and is therefore redundant. Furthermore it allows the system to be scale-invariant as all hands are reduced to the same size. Once all of this has been done, the shape distribution technique designed by [10] is then used to generate a histogram for each of the 26 training images. This is done by performing the following steps:

1. Select a random point on the hand
2. Select another random point on the hand
3. Use Pythagoras' theorem to calculate the distance between the two random points
4. Increment the value of the bin on the histogram that corresponds to this distance

This process is repeated a specified number of times and a histogram is produced. The number of calculations that are performed would depend on the purpose of the system. See Table 1 for the correlation between the amount of calculations and the accuracy and time it takes to perform. Each histogram produced is then stored and can be referenced as a comparison operand. As each histogram is composed of random values every time it is instantiated, it is highly unlikely that two histograms will ever be *exactly* the same. However, given that a particular shape will produce a histogram of a certain signature, it can be resolved that the next time the algorithm is run for said shape, the signature, although not identical, will be very comparable - more so than a signature for another shape.

A histogram is then produced for every letter of the sign language alphabet. These 26 images are then used as the templates for comparison. Once this is done, it is then possible to input another image of a hand performing a posture of the alphabet in sign language. Some letters, their corresponding input images in sign language, and their resulting histogram are illustrated in Figure 2. In each distribution, the horizontal axis represents distance, and the vertical axis represents the probability of that distance between two points on the surface.

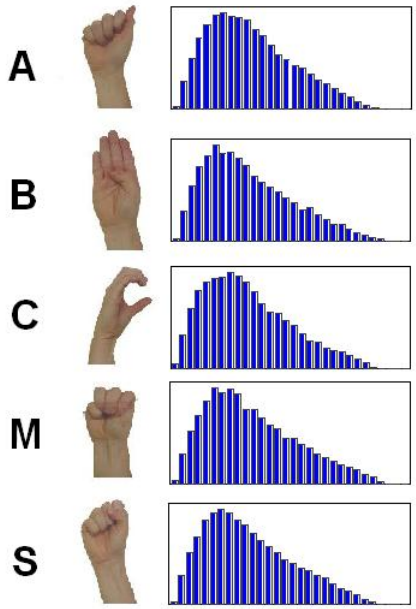

Figure 2: Sample shape distributions.

A histogram is then produced for this image using the same steps as described above. This histogram is then compared against the 26 histograms relating to the stored images of the alphabet. The Earth Mover's Distance formula has been implemented to perform this graph comparison task [12]. Intuitively, given two histograms, one can be seen as a mass of earth properly spread in space, the other as a collection of holes in that same space. The EMD then measures the least amount of work needed to be done to fill the holes with earth. Computing the EMD then becomes a solution to the well-known "transportation problem"[6]. The least amount of work required to fill the holes with earth equates to a figure. We check that this figure is less than a set threshold to determine how comparable the input image is to the training image.

4 Experiments and Results.

The aforementioned process has been coded completely in Java. This includes all image acquisition and processing techniques as well as the shape distribution method and graph comparison operation. The system

comprises of a number of JPEG's stored locally on the machine and a standalone Java application. Each JPEG is between 20KB and 25KB in size, taken using a regular 6.6 Megapixel digital camera. Once some standard image pre-processing techniques have been performed on all of the images, the Java program then retrieves the images and performs the recognition process. When a posture has been correctly identified, a counter is incremented. This counter is then used at the end of the program to calculate a percentage of accurate matches.

Initial tests on the system used two sets of 26 images. One set is used to establish templates, while the other is used for input parameters. The hands in the images were of different users and taken under different lighting conditions and different distances from the camera, thus illustrating a level of robustness. The system has primarily been developed as a pre-classification stage with the overall goal of achieving total hand posture recognition. By implementing this simple step, it is possible to reduce the number of images to be recognised by a considerable amount. In other words, by analysing an input image and performing the comparison technique, the system can filter the number of potential matches to only a few.

The success rate of the system depends greatly on the number of random distance calculations that are performed. If we take a high number of calculations, it produces a very specific histogram. This then allows for a higher element of accuracy at the comparison stage. However, by increasing the number of calculations that must be performed, the length of time the process takes also increases. As we can see from Table 1, there is a linear trade-off between accuracy and time. The time displayed in the table includes the time taken to train the system, which also increases in a linear fashion. The values in the table represent the time and accuracy produced for comparing 250 input images. These input images are compared against the 26 images that have been prepared off-line. In Table 1, the system narrowed the set of images from 26 down to 6. The percentage value represents whether the input image is an element of the reduced set.

Table 1: Accuracy rates for different calculations

# Calculations	Time	Accuracy
5,000	2 min. 51 secs.	65%
15,000	3 min. 1 sec.	71%
25,000	3 min. 30 secs.	75%
50,000	4 mins.	80%
150,000	5 mins. 42 secs.	89%

From the table it is easy to understand how the number of calculations taken has a direct effect on the accuracy of the process. However, it is worth noting that given a large amount of calculations, the system is successful in matching the input images - even distinguishing between postures that are very similar in appearance.

5 Conclusions and Future Work.

Through the course of the research presented in this paper, a hand posture recognition pre-classification system has been developed by implementing an inherently simple shape distribution technique. There is room for improvement in this system at the recognition stage. Perhaps a more accurate graph comparison operation might improve the overall result.

Furthermore, the system performs on a two-dimensional basis. Recent research has suggested that a thorough sign language recognition system would require three-dimensional input in order to completely capture the gesture correctly [15] . The research undertaken for this paper did not present a scenario whereby a *posture* requires three-dimensional input. However, it is possible that self-occlusion may occur on the posture as it moves through a gesture. If this were the case, this system would require a progression to three dimensions. Presently, however, it would only be necessary to incorporate this two-dimensional posture recognition system into a three-dimensional gesture recognition system. This task would then prove whether a development from 2D to 3D would be necessary for this system.

As was mentioned earlier, the input images consist of the hand, wrist and background. In a real-world scenario, it is likely that the input image would be a person's full body - as gestures would also need to be captured. In this case, it would be necessary to segment the image so that only the hand performing the posture is input to this system. Such segmentation techniques have been developed [15], but have not been implemented in this system yet.

Rather than using this algorithm as an end-goal posture identification system, it has been used as a pre-classification step in said area. That is to say, if a highly accurate posture identification system were developed, but its performance was slow, the system presented in this paper could be used to narrow down the number of images to be analysed. Thus allowing for the more accurate system to be implemented on fewer images.

References

[1] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. *Internation Joint Conferences on Artificial Intelligence*, pages 659–663, 1997.

[2] Stefan Canzar and Jan Remy. Shape distributions and protein similarity. 2003.

[3] Feng-Sheng Chen, Chih-Ming Fu, and Chung-Lin Huang. Hand gesture recognition using a real-time tracking method and hidden markov models. *Image and Vision Computing*, 21:745–758, March 2003.

[4] Q. Delamarre and O. Faugeras. Finding pose of hand in video images: A stereo-based approach. In *IEEE Proceedings of the Third International Conference on Automatic Face and Gesture Recognition*, pages 585–590, Washington, DC, USA, 1998.

[5] N. Diakopoulous. A curve matching approach to gesture recognition. In *www.lems.brown.edu/ nad/.*

[6] F. L. Hitchcock. The distribution of a product from several sources to numerous localities. *Journal of Mathematical Physics*, 20:224–230, 1941.

[7] Pengyu Hong, Matthew Turk, and Thomas Huang. Gesture modelling and recognition using finite state machines. In Pengyu Hong, Matthew Turk, and Thomas Huang, editors, *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, Grenoble, France, 28–30 March 2000.

[8] Rung-Huei Liang and Ming Ouhyoung. A real-time continuous gesture recognition system for sign language. In *Third international conference on automatic face and gesture recognition*, pages 558–567, April 1998.

[9] Andrew Litvin and William Clem Karl. Using shape distributions as in a curve evolution framework. California, May 2004. IS&T/SPIE, Annual Symposium Electronic Imaging Science and Technology.

[10] Robert Osada, Thomas Funkhouser, Vernard Chazelle, and David Dobkin. Matching 3d models with shape distributions. *ACM Transactions on Graphics*, 21(4):807–832, October 2002.

[11] James M. Rehg and Takeo Kanade. Model-based tracking of self-occluding articulated objects. In *Proceedings of 5th International Conference on Computer Vision*, pages 612–617, Cambridge, MA, June 1995.

[12] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, November 2000.

[13] Jung Soh, Ho-Sub Yoon, and Min Wang. Locating hands in complex images using colour analysis. In *International conference on systems, man, and cybernetics*, pages 2142–2146, October 1997.

[14] Gabriel Somers and Nigel Whyte. Identifying hand movement patterns in three dimensions using finite state machines. In Gabriel Somers and Nigel Whyte, editors, *Proceedings of the 12th Irish Conference on Artificial Intelligence and Cognitive Science*, pages 297–306, Maynooth, Ireland, 2001.

[15] Gabriel Somers and Nigel Whyte. Hand posture matching for irish sign language interpretation. In *ACM Proceedings of the 1st International Symposium on Information and Communication Technologies*, pages 439–444, Trinity College Dublin, Ireland, September 24–26 2003. Trinity College Dublin.

[16] E. Stergiopoulou, N. Papamarkos, and A. Atsalakis. Hand gesture recognition via a new self-organized neural network. Havana, Cuba, November 2005.

[17] E. Ueda, Y. Matsumoto, M. Imai, and T. Ogasawara. Hand pose estimation using multi-viewpoint silhouette images. In *2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1989–1996, 2001.

[18] Z. Zhang. Iterative point matching for registration of free-form curves. In *Rapport de Recherche I.N.R.I.A. Numero 1658*, March 1992.