A Survey on Sign Language Recognition Using Smartphones

Sakher Ghanem[1,2] , Christopher Conly[1], Vassilis Athitsos[1]
[1]Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, Texas, USA
[2]Faculty of Computing and Information Technology, University of Jeddah, Jeddah, Saudi Arabia
sghanem@uj.edu.sa, chris.conly@uta.edu, athitsos@uta.edu

ABSTRACT

Deaf people around the globe use sign languages for their communication needs. Innovations of new technologies, such as smartphones, offer a host of new functionalities to their users. If such mobile devices become capable of recognizing sign languages, this will open up the opportunity for offering significantly more user-friendly mobile apps to sign language users. However, in order to achieve satisfactory results, there are many challenges that must be considered and overcome, such as light conditions, background noise, processing, and energy limitations. This paper aims to cover the most recent techniques in mobile-based sign language recognition systems. We categorize existing solutions into sensors-based and vision-based, as these two categories offer distinct advantages and disadvantages. The primary focus of this literature review is on two main aspects of sign language recognition: feature detection and sign classification algorithms.

CCS Concepts

•Computing methodologies → Computer vision; Object detection; Object recognition;

Keywords

Sign Language Recognition; Smartphone; Portable Device; Survey

1. INTRODUCTION

According to a report from Gallaudet University, which is a prominent educational institution that serves people who are deaf or are hard of hearing, there are approximately 38 million deaf individuals in the United States [8]. Many of those individuals use a sign language, typically American Sign Language (ASL), as a primary or secondary form of communication. Sign languages (SLs) are necessarily visual in nature. For sign language users, communicating with hearing people can be a challenge. Similarly, important information technology and social connectivity tools are not available to sign language users, unless the users are willing to access such tools using a spoken and written language, such as English, with which they may not be comfortable. Technological innovations in automated sign recognition have the potential to help sign language users overcome such obstacles, by facilitating both communication with hearing people, and human-computer interaction.

Mobile computing has entered a new era where mobile phones are powerful enough to be used in such advanced applications as gesture and sign language recognition. Many of the newly designed smartphones are equipped with multicore processors, a high-quality GPU, and a high-resolution camera that can reach 12MP and more. These high-tech features allow the devices to execute computationally intensive tasks in less amount of time. In the last decade, many applications of computer vision have been limited to desktops, and now with the availability of advanced processor-equipped smartphones, computer vision is primed to experience a transformation to provide new experiences via mobile devices.

Research has shown that ASL has four basic manual components: finger configuration of the hands, movement of the hands, orientation of the hands and the location of the hands with respect to the body [3]. Any automated sign recognition system needs two main procedures: the detection of the features and the classification of the input data. With mobile phones, the detection process can be affected by the movement of the phone, which causes extraneous motion around the signer. Some techniques use a sensor-based technology which tracks the gestures via hand movement using embedded sensors. Other techniques utilize vision-based approaches to process images of the captured gesture. Also, several researchers suggest using a client-server architecture to speed up processing time.

This literature review covers existing sign language recognition systems designed to run on smartphones. The lack of a clear overview in this area is the primary motivation to present this work. This survey presents several existing methods and groups them in different categories. The methods are discussed with a focus on the feature detection and classification algorithms.

The rest of the paper is organized as follows. Section 2 discusses the datasets used in this area. Section 3 describes existing approaches for sign language recognition in portable devices, including sensor-based and vision-based

PETRA '17, June 21-23, 2017, Island of Rhodes, Greece
 ISBN 978-1-4503-5227-7/17/06…$15.00
DOI: http://dx.doi.org/10.1145/3056540.3056549

Figure 1: ASL signs representing numbers 0-9 and letters of the English alphabet. [32]

approaches. Finally, conclusions and possible future directions of the technology are discussed in Section 4.

2. SIGN DATASETS

In general, there are two types of signs: dynamic and static. Dynamic signs exhibit motion, whereas static signs are characterized by a specific static posture. We did not find any dataset that was designed exclusively for sign language application in portable devices. Some researchers use a static set of gestures, capturing signs for letters of the English alphabet and numbers 0-9, e.g., [26]. Figure 1 depicts American Sign Language signs representing numbers and letters. In many implemented methods, a customized dynamic dataset is utilized, e.g., [14]. It is difficult to handle the available datasets that were designed for personal computers due to the limited storage capacity of mobile phones.

3. SIGN LANGUAGE RECOGNITION USING SMARTPHONES

In sign language recognition, the motion and posture of the human hand can observed via different approaches. In the sensor-based approach, the movement of the hand is tracked via sensors attached to wireless gloves or sensors embedded in smartphones, and appropriate techniques are used to process the responses from the sensors. In the vision-based approach, the gestures are observed via a mobile camera, and multiple processing steps are applied to recognize the signs that appear in the video stream.

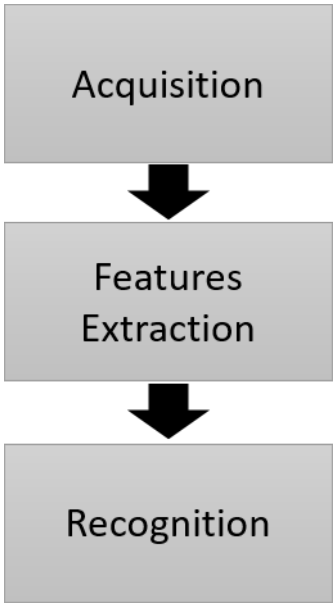

Figure 2: Basic System Architecture.

Any sign recognition system contains three major steps; see Figure 2 for an overview. First, the input data is acquired, for example via the phone camera or from some sensor. The next step is to extract the features from the input data. Finally, the sign is classified using some appropriate algorithm that is compatible with the extracted features. For each method we examine, we take a close look in how that method approaches the problems of feature extraction and recognition/classification.

3.1 Sensor-Based Approach

The usage of sensors simplifies the detection process and makes it faster. At the same time, sensor-based systems can be expensive and cumbersome to use, and these factors discourage adoption by a large number of users. Table 1 demonstrates a comparison between existing sensor-based models that use the phone as a platform. Sensor-based approaches can be broadly categorized based on whether they use external sensors, such as gloves, or internal sensors built into the smartphone. The following two subsections discuss these two categories.

3.1.1 Using Gloves

Glove-based approaches have been implemented using sensors that track hand gestures. Multiple sensors embedded in the gloves are used to track the fingers, palm and their location and motion. Such an approach provides coordinates of the palm and fingers for further processing. These devices may be connected wirelessly via Bluetooth.

The detection of hand parameters in this approach relies on a customized glove [14, 20] that contains ten flex sensors to track the posture of each finger. Moreover, a G-sensor is used to monitor the orientation of the hand. Hand motion is

Table 1: A Comparison of Available Sensors Based Systems

System	Sensors	Classification Method	Gesture Type	Processing	Voc. Size	Dependency
Kau 2015 [14]	Gloves	Template Matching	Dynamic	Local	5	user-independent
Preetham 2013 [20]	Gloves	Minimum Mean Square Error Algorithm	Static	Local	-	-
Seymour 2015 [26]	Gloves	SVM	Static	Local	31	user-independent
Choe 2010 [2]	Phone Internal Sensors	DTW	Dynamic	Local	20	user-independent
Gupta 2016 [6]	Phone Internal Sensors	DTW	Dynamic	Local	6	user-independent
Joselli 2009 [11]	Phone Internal Sensors	HMM+foreword -backward algorithm	Dynamic	Local	10	user-independent
Niezen 2008 [17]	Phone Internal Sensors	DTW	Dynamic	Local	8	user-independent
Wang 2012 [29]	Phone Internal Sensors	own statical method	Dynamic	Local	21	user-dependent

detected by using a gyroscope sensor that calculates angles of the hands in space. These sensors continuously trace the signal to get hand data. The data are transferred wirelessly to the mobile device. From the gathered data, the state of the hand is estimated. This state can be decomposed into four independent components: hand posture, position, orientation, and motion.

The recognition methods vary by the available input data and the dataset used. Template matching was used in [14] as a classification method using five dynamic sign classes. In [26], a comparison is made between SVM and neural network methods using two different activation functions: log-sigmoid and symmetric Elliott functions. The experiment was done using static hand gestures, representing letters and numbers. In the results, SVM produced better accuracy, but it required 16 times more time for classification, compared to Log-sigmoid neural network and symmetric Elliott neural network. The advantage of this method was memory usage: only 4 MB of memory were required, which makes this method usable even with low-end smartphones.

3.1.2 Smartphone Internal Sensors Approach

Recently, new smartphones have been embedded with sensors that help to detect the posture and motion of the device. Numerous researchers utilize this feature to create gesture recognition models. The main issue with this approach is the limitation of signs details provided by the sensors.

Gestures recognized using such sensors can be decomposed into sequences of two simpler gesture types [29]. Turn gestures correspond to a change in the 3D orientation of the device. An example is rotating the device from the face up to face down position. Translation gestures correspond to the phone moving in 3D space. Moving the phone up and down is an example of such a gesture. Segmentation of the motion is performed to detect the start and end point of the movement. Since the accelerometer continuously reads data of the three axis point in space, a vector containing the sum of derivatives of the current axis with the previous axis can be used to detect motion, as done in [11, 17, 29, 2]. To speed up the calculation time, **Gupta 2016** [6] change mean floating values to integer values by using a probability density function.

One of the better-known classification methods is the Dynamic Time Warping (DTW) algorithm, which is applied to measure the cost of a selected gesture compared with training data [28, 15]. One of the main advantages of this algorithm is that it does not need large amounts of training data, as it can be used even when only a single training example per class is available. DTW is used by [17, 6, 2] to achieve high accuracy, under the assumption that the start and end times for every sign are known. **Joselli 2009** [11] adapted foreword-backward algorithm to classify dynamic input signs using Hidden Markov Models (HMM), and using a database containing ten classes with a total of 400 samples. **Wang 2012** [29] process the data from the sensors to develop a sinusoid-like curve that can be used to extract the pattern of the movement. The axis of the largest variance between peak and valley is the movement direction.

3.2 Vision Based Approach

In recent years, the availability and simplicity of smartphones has encouraged researchers to utilize them in vision-based sign language recognition applications. The vision-based approach uses the phone camera to capture the image or the video of the hand performing signs. These frames are further processed to recognize the signs, so as to produce text or speech output. Vision-based approaches risk producing relatively low accuracy compared to sensor-based approaches, due to multiple challenges in image processing, like light variations, dependency on the skin color of the user, complex backgrounds in the image, etc. Table 2 shows a comparison between currently existing vision based methods. It is important to note that all approaches listed in this table use static signs, except **Rao 2016** [22] which includes dynamic signs.

Extracting accurate hand features is a major challenge for the vision-based approach. Extraction is affected by many factors, such as lighting condition and background noise. The more accurate the detection and extraction is, the better the recognition results become. Orientation and position of the hand can be detected in different ways, for example using skin detection or Viola-Jones cascades of boosted rect-

Table 2: A Comparison of Available Vision Based Systems

System	Features Extraction	Classification Method	Processing	Voc. Size	Dependency
Elleuch 2015 [4]	Skin detection HSV, convexity defects	SVM	Local	5	user-independent
Gandhi 2015 [5]	Background subtraction	Template matching	Local	-	-
Hakkun 2015 [7]	Viola-Jones Haar Filters	KNN	Local	8	user-dependent
Hays 2013 [9]	Skin detection YCrCb, canny edge	SVM	Local, Client-Server	32	user-independent
Jin 2016 [10]	Skin detection RGB, canny edge, SURF	SVM	Local	16	user-dependent
Joshi 2015 [12]	PCA	SVM	Local	5	user-independent
Kamat 2016 [13]	Skin detection RGB	Template Matching	Local	4	user-dependent
Lahiani 2015 [16]	Skin detection RGB, convexity defects	SVM	Local	10	user-dependent
Prasuhn 2014 [19]	Skin detection HUV, HOG	Brute-force Matching	Client-Server	26	user-dependent
Raheja 2015 [21]	Sobel Edge Filter, PCA	Template Matching	Client-Server	10	user-dependent
Rao 2016 [22]	Gaussian and Sobel Edge Filter + PCA	MDC	Local	18	user-independent
Saxena1 2014 [24]	Sobel Edge Filter	Backpropagation Algorithm	Client-Server	5	user-dependent
Saxena2 2014 [25]	Skin detection RGB, PCA	Template Matching	Client-Server	10	user-dependent
Warrier 2016 [30]	Skin detection RGB	Geometric Matching	Client-Server	11	user-dependent

angle filters [27]. Detecting the position and orientation of the hand at each frame accurately also allows us to detect the motion of the hand for dynamic signs.

Skin segmentation algorithms, which often depend on specifying thresholds [18], are widely used in Computer Vision applications. The researchers either specify skin thresholds manually or automatically by taking a skin color sample before the experiment. Several available models use RGB color space, e.g., [10, 13, 16, 25, 30]. To solve brightness and lighting problems, [9] use YCrCb color space, [4] employ HSV color space, and [19] benefit from HUV color space.

The Viola-Jones detection method [27], which uses cascades of boosted rectangle filters, is a well-known method, that is commonly used for detecting hands. Some researchers [7, 4] implement the Viola-Jones method on portable platforms, as Viola-Jones is relatively easy to implement and has low hardware requirements. Another alternative, used by [12, 21, 22, 25], is Principal Component Analysis (PCA).

Additional hand details are also extracted by various methods. Examples of such details are the number of open fingers (measured by finding contours), finding the palm area (by finding the largest circle that fits in the hand region), detecting the convex hull, and getting convexity defects [4, 9, 16]. Canny edge detection [1] can also be used to identify the hand area [10]. Likewise, a Sobel Edge filter, which measures the changes in value in the highest moving direction, has been used [21, 22, 24]. **Prasuhn 2014** [19] apply a Histogram of Orientation Gradients (HOG) method, which is sensitive to the angle of the object, to extract the features from the input image. Another method, used by [5], is background subtraction using a motion detection method. In **Jin 2016** [10], Speeded-Up Robust Features (SURF) is used as an extra feature to improve accuracy.

Once the features describing a sign have been extracted, there are numerous recognition procedures that can be applied. Support Vector Machines (SVM) define decision boundaries between classes, which are linear in some transformed feature space, but can be highly nonlinear in the original feature space [31]. Several papers use SVMs, e.g., [4, 9, 10, 12, 16]. **Hakkun 2015** [7] use K-Nearest Neighbor (KNN) for classification. Another simple technique for classification is template matching, used by [5, 13, 30, 21, 12]. The Backpropagation algorithm [23] can lead to very efficient classification timewise, but it needs more training data to minimize error rate. Backpropagation is used by [24] as the recognition method. In **Rao 2016** [22], because the speed of processing in portable devices is a major factor, a minimum distance classifier (MDC) was chosen as a classification method. The experiments use sentences of signs as training and test data.

Some systems assume that the only visible object in the captured image is the hand [7, 4], while the more advanced models manage to capture both hands and face. One way to remove the confusion between a face and hand area is to subtract or isolate the face, so that the detection of hand details will be more precise [4]. Another issue that can be considered is hand angle and hand distance from the mobile device. In tests conducted in [7], optimal results were achieved with no more than 50 cm distance between the hand and the camera, and the hand being in the upright state.

Due to slow processing time in some models, a client-server framework is used. In such a framework, the phone is connected to a regular computer via wireless network. Such an approach was implemented in [19, 21, 24, 25, 30]. A cloud service can be used to execute part of recognition operations, as done in [9]. Moreover, **Elleuch 2015** [4] implement a multithreading technique by running face subtraction and hand pre-processing at the same time, thus decreasing the processing time by half.

4. CONCLUSIONS

In this paper, we have provided a survey of existing techniques for sign language recognition in smartphones. We discussed sensor-based approaches, which track hand motion and/or posture using hardware-based trackers installed in a glove or inside a smartphone. We also discussed vision-based approaches, which use the phone camera for observing the hand. In discussing both types of approaches, we focused on the detection and feature extraction module as well as the classification module of each approach.

Regarding vision-based methods, significant challenges remain to be overcome by future research, regarding accuracy of hand detection and articulated hand pose estimation, as well as classification accuracy. Most existing vision-based methods only recognize static gestures, and we expect new methods to be proposed for handling dynamic gestures. Similarly, existing methods typically cover no more than a few tens of signs, and there is significant room for improvement until methods can cover the several thousands of signs that users of a sign language employ in their daily usage. Extending vision-based recognition systems to cover dynamic gestures and thousands of signs may strain the hardware capabilities of smartphones. While smartphone hardware specs are expected to continue to improve rapidly, cloud processing could push the boundaries further ahead by alleviating the hardware requirements on the mobile device. However, maintaining interactivity and low latency while using cloud processing can also be challenging, and these are also issues that we expect future research to focus on.

5. ACKNOWLEDGMENTS

This work was partially supported by National Science Foundation grants IIS-1055062 and IIS-1565328. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors, and do not necessarily reflect the views of the National Science Foundation.

6. REFERENCES

[1] J. Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986.

[2] B. Choe, J.-K. Min, and S.-B. Cho. Online gesture recognition for user interface on accelerometer built-in mobile phones. In *International Conference on Neural Information Processing*, pages 650–657. Springer, 2010.

[3] E. Costello. *American sign language dictionary.* Random House Reference &, 2008.

[4] H. Elleuch, A. Wali, A. Samet, and A. M. Alimi. A static hand gesture recognition system for real time mobile device monitoring. In *Intelligent Systems Design and Applications (ISDA), 2015 15th International Conference on*, pages 195–200. IEEE, 2015.

[5] P. Gandhi, D. Dalvi, P. Gaikwad, and S. Khode. Image based sign language recognition on android. *International Journal of Engineering and Techniques*, 1(5):55–60, 2015.

[6] H. P. Gupta, H. S. Chudgar, S. Mukherjee, T. Dutta, and K. Sharma. A continuous hand gestures recognition technique for human-machine interaction using accelerometer and gyroscope sensors. *IEEE Sensors Journal*, 16(16):6425–6432, 2016.

[7] R. Y. Hakkun, A. Baharuddin, et al. Sign language learning based on android for deaf and speech impaired people. In *Electronics Symposium (IES), 2015 International*, pages 114–117. IEEE, 2015.

[8] S. Hamrick, L. Jacobi, P. Oberholtzer, E. Henry, and J. Smith. Libguides. deaf statistics. deaf population of the us. *Montana*, 16(616,796):2–7, 2010.

[9] P. Hays, R. Ptucha, and R. Melton. Mobile device to cloud co-processing of asl finger spelling to text conversion. In *Image Processing Workshop (WNYIPW), 2013 IEEE Western New York*, pages 39–43. IEEE, 2013.

[10] C. M. Jin, Z. Omar, and M. H. Jaward. A mobile application of american sign language translation via image processing algorithms. In *Region 10 Symposium (TENSYMP), 2016 IEEE*, pages 104–109. IEEE, 2016.

[11] M. Joselli and E. Clua. grmobile: A framework for touch and accelerometer gesture recognition for mobile games. In *2009 VIII Brazilian Symposium on Games and Digital Entertainment*, pages 141–150. IEEE, 2009.

[12] T. J. Joshi, S. Kumar, N. Tarapore, and V. Mohile. Static hand gesture recognition using an android device. *International Journal of Computer Applications*, 120(21), 2015.

[13] R. Kamat, A. Danoji, A. Dhage, P. Puranik, and S. Sengupta. Monvoix-an android application for hearing impaired people. *Journal of Communications Technology, Electronics and Computer Science*, 8:24–28, 2016.

[14] L.-J. Kau, W.-L. Su, P.-J. Yu, and S.-J. Wei. A real-time portable sign language translation system. In *2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 1–4. IEEE, 2015.

[15] J. Kruskall and M. Liberman. The symmetric time warping algorithm: From continuous to discrete. time warps, string edits and macromolecules, 1983.

[16] H. Lahiani, M. Elleuch, and M. Kherallah. Real time hand gesture recognition system for android devices. In *Intelligent Systems Design and Applications (ISDA), 2015 15th International Conference on*, pages 591–596. IEEE, 2015.

[17] G. Niezen and G. P. Hancke. Gesture recognition as ubiquitous input for mobile phones. In *International Workshop on Devices that Alter Perception (DAP 2008), in conjunction with Ubicomp*, pages 17–21. Citeseer, 2008.

[18] S. L. Phung, A. Bouzerdoum, and D. Chai. Skin segmentation using color pixel classification: analysis and comparison. *IEEE transactions on pattern analysis and machine intelligence*, 27(1):148–154, 2005.

[19] L. Prasuhn, Y. Oyamada, Y. Mochizuki, and H. Ishikawa. A hog-based hand gesture recognition system on a mobile device. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 3973–3977. IEEE, 2014.

[20] C. Preetham, G. Ramakrishnan, S. Kumar, A. Tamse, and N. Krishnapura. Hand talk-implementation of a gesture recognizing glove. In *India Educators' Conference (TIIEC), 2013 Texas Instruments*, pages

328–331. IEEE, 2013.

[21] J. L. Raheja, A. Singhal, and A. Chaudhary. Android based portable hand sign recognition system. *arXiv preprint arXiv:1503.03614*, 2015.

[22] G. A. Rao and P. Kishore. Sign language recognition system simulated for video captured with smart phone front camera. *International Journal of Electrical and Computer Engineering (IJECE)*, 6(5):2176–2187, 2016.

[23] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

[24] A. Saxena, D. K. Jain, and A. Singhal. Hand gesture recognition using an android device. In *Communication Systems and Network Technologies (CSNT), 2014 Fourth International Conference on*, pages 819–822. IEEE, 2014.

[25] A. Saxena, D. K. Jain, and A. Singhal. Sign language recognition using principal component analysis. In *Communication Systems and Network Technologies (CSNT), 2014 Fourth International Conference on*, pages 810–813. IEEE, 2014.

[26] M. Seymour and M. Tšoeu. A mobile application for south african sign language (sasl) recognition. In *AFRICON, 2015*, pages 1–5. IEEE, 2015.

[27] P. Viola and M. J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[28] H. Wang, A. Stefan, S. Moradi, V. Athitsos, C. Neidle, and F. Kamangar. A system for large vocabulary sign search. In *European Conference on Computer Vision*, pages 342–353. Springer, 2010.

[29] X. Wang, P. Tarrío, E. Metola, A. M. Bernardos, and J. R. Casar. Gesture recognition using mobile phone's inertial sensors. In *Distributed Computing and Artificial Intelligence*, pages 173–184. Springer, 2012.

[30] K. S. Warrier, J. K. Sahu, H. Halder, R. Koradiya, and V. K. Raj. Software based sign language converter. In *Communication and Signal Processing (ICCSP), 2016 International Conference on*, pages 1777–1780. IEEE, 2016.

[31] J. Weston and C. Watkins. Multi-class support vector machines. Technical report, Citeseer, 1998.

[32] Wikipedia. American manual alphabet, https://en.wikipedia.org/wiki/american_manual_alphabet, 2016.