Journal of Engg. Research **Vol. 4 No. (1) March 2016** pp. 21-41

التعرف على لغة الإيماءات الباكستانية باستخدام الصور

محمد رئيس، صحت الله، سميع الرحمان واحسان ربى
قسم هندسة الكمبيوتر وعلوم تكنولوجيا، جامعته مالاكند، باكستان

الخلاصة

اللغة الرمزية هي لغة الايماءات المستخدمة للتواصل غير اللفظي. هذة الورقة تبحث التعرف على الأحرف الهجائية والعلامات الرقمية في لغة الإشارة الباكستانية. تم إستقصاء التحليل المبني على البكسل المتعمق للتعرف على الأصابع (من الفهرست إلى الخنصر) بينما وضع الإبهام تم التعرف عليه من خلال مطابقة القالب. بعد التوقف على الأصابع يتم التعرف على موضع الأصابع (مرفوع – مطوي) بالمقارنة مع وضع الإبهام بواسطة الأبعاد الثنائية. للتعرف السريع تم تصنيف الإشارات في سبع مجموعات الخوارزمية المستخدمة تتعرف على هذه المجموعات السبع. دقة الطريقة بلغت مستوى مرضي (84.2%) عندما استخدمت 180 رقم و240 حرف إشاراتي.

Image based recognition of Pakistan sign language

Muhammad Raees, Sehat Ullah, Sami Ur Rahman and Ihsan Rabbi*

Department of Computer Science and IT, University of Malakand, Pakistan

**Corresponding author: ihsanrabbi@uom.edu.pk*

ABSTRACT

Sign language is the language of gestures used for non-verbal communication. This paper deals with alphabets and digit signs recognition from Pakistan Sign Language (PSL). The deep pixels-based analysis is pursued for the recognition of fingers (from index to small finger) while thumb position is determined through Template Matching. After fingers identification, the isolated signs are recognized based on finger states of being raised or lowered besides thumb's position in 2D. For a quick recognition, signs are categorized into seven groups. The algorithm identifies these groups following a model of seven phases. The system's accuracy achieved a satisfactory level of 84.2% when evaluated with signs comprising 180 digits and 240 alphabets.

Keywords: Alphabets signs recognition; digits signs recognition; fingers detection; hand gestures recognition; Pakistan sign language.

INTRODUCTION

Sign Language (SL) is used to communicate through gestures instead of sounds. Like spoken languages, sign language is natural because mute people learn it as their mother tongue imperceptibly. Although there is no international sign language, every region of the world has its own sign language. Like American Sign language (ASL) and British Sign Language (BSL), Pakistan has its own sign language called Pakistan Sign Language (PSL) where signs are used to represent Urdu letters and terms. Mute people always need an interpreter for their sign language to enable them to communicate in the society for their day to day dealings. Let alone the availability of an interpreter, it is humanly impossible for an interpreter to accompany an individual (mute) all the time and assist him in every matter of life. This paper is the first of our long-term goal of designing an Automated Urdu Sign Language Interpreter (AUSLI) to help mute people.

Unlike other sign language systems, which rely on prior training (Subha & Balakrishnan, 2011), the proposed algorithm dynamically determines each types of finger i.e. which finger is small, which one is middle and so on. Keeping in view the position of each known finger, the system recognizes all those isolated signs, represented by the palm side of hand, by identifying whether the fingers are open or close. The two earlier efforts made for Urdu alphabets recognition of PSL are the approaches of Alvi et *al.,* (2005) and Sumera *et al.*, (2008). The first one is 5DT Data Glove based which compares a scanned gesture image with the images stored in the database. The second PSL recognition system needs the glove of eleven different colours. However, both the systems recognize only alphabets of PSL.

The existing approaches for sign language utilize either Data Gloves or Computer Vision. The former is not only expensive but also needs cumbersome set-up of wires. The latter approach may be classified into two categories. The first category algorithms are based on the latest Kinect system. The accuracy of Kinect based systems is comparatively high but the system is expensive. The second category is a camera based system which is less expensive but subject to light sensitivity.

Our designed system, though camera based, tolerates a moderate change of light intensity. Furthermore, the distance from camera problem has also been successfully resolved by calculating finger width before sign identification. One distinguishing feature of the proposed approach is that neither colours nor gloves are used for finger identification. To further reduce computation, instead of scanning the whole scene, the system captures images only from a specified Region of Interest (ROI). While displaying the appropriate Urdu digit/alphabet sign, the system also pronounces it on behalf of the signer.

LITERATURE REVIEW

Apart from Sign Language (SL), hand gestures are commonly used in different Human Computer Interaction (HCI) domains like traffic controllers, robot instructions, image navigation, hand-free computing and remote video gaming. The crux of the matter is to enable the computer to recognize hand gestures. Several methods have been suggested to detect the signs easily and accurately. Among the noteworthy methodologies used for sign recognition are Template Matching (Liu & Fujimura, 2004), Conditional Random Fields (CRF) (Saad *et al.*, 2012) and Dynamic Time Warping (DTW) (Doliotis *et al.*, 2011). Dynamic signs are the signs which rely on hands, head and body motion. The procedure proposed by Fan & Ling (2014) for human actions and events recognition, is pertinent to be followed for Dynamic signs recognition systems. The methodology of Transition-Movement Models (TMM) is proposed in Fang *et al.* (2007) where movements of hands are recognized to deduce continuous signs. Although the system recognizes a large vocabulary of 513 signs, the technique is rarely affordable due to its use of costlier sensor based glove with magnetic trackers.

Hidden Markov Model based hand gesture recognition proposed by Wen *et al.* (2000) achieved 82.2% accuracy in recognition of 20 Arabic alphabet signs. The system needs pre-training of hand gestures in which an input image sign is compared by using eight feature points. The View-Desk algorithm of Starner *et al.* (1998) uses a 40-word lexicon with 90% accuracy but suffers from a slight unintentional head or body movement. Most of the sign recognition systems are glove based because finger positions and orientations are easily detectable through it. The system proposed by Al-Jarrah & Alaa (2001) uses Cyber Gloves while that of Kadous (1996) needs Power Gloves. The approach based on Data Gloves for FSL (French Sign Language) Yang & Lee (2010) has 83% accuracy. Although all signs of Arabic alphabets can be recognized by the algorithm of Assaleh & Al-Rousan (2005), however, due to light sensitivity, its accuracy remains below 57%. The algorithm of Simon *et al.* (2011) which makes use of kinect with infrared camera, is independent of lighting conditions. As some signs need facial expression and body motion, *kinect* based system could hardly cope with such dynamic signs. Skin colour segmentation is followed in Vezhnevets & Reeva (2003) and Zabulis *et al.* (2007) where hand section is extracted from the image. Boundary tracking and fingertip methodology is pursued for sign detection in Ravikiran *et al.* (2009). Though the algorithm is robust as only 5% errors occurred in the detection of all those signs of ASL in which fingers are open, the approach is neither applicable for signs in which fingers are closed nor for the motion based signs. The method of thinning the segmented image is presented in Rajeshree & Manesh (2009).

The Object To Class (O2C) distance for modelling object bank proposed by Zhang *et al.* (2014), is a significant model for features-extraction based visual sign language recognition. The technique of Morphological operation has been exercised in Kuch & Huang (1995) for hand gestures recognition. The work of Nagarajan *et al.* (2012) identifies hand postures used for signs of counting with the help of Convexity defects. Convexity defects of fingers vary to a great extent with a slight orientation of hand or of fingers. The Non-manual features extraction is a sub-domain of sign language used for continuous signs. Head pose and motion for sign recognition is suggested in Erdem & Sclaroff (2002) where head shakes based signs are well recognized by trackers. Facial expression and Gaze direction are traced in the approaches discussed in Von *et al.* (2008) and Muir & Leaper (2003), respectively, for continuous signs. The module of Shanableh (2007) extracts features by *K-NN* and Polynomial Networks. The Visual Command system is suggested by Chalechale & Safae (2008) where hand posture is surrounded by ordinary daily items. The work is interesting and can pave the way in the domain for a new sign language.

Image classification contributes significantly to the domain of sign language recognition. The evolutionary learning methodology of Ling *et al.* (2014) generates domain-adaptive global feature descriptors for image classification and may be pursued further for automated sign recognition.

Till date, the literature of sign languages has no approach proposed especially for PSL digits or numbers. This paper is the first to present a novel algorithm for recognition of both alphabets and digits of PSL. The algorithm neither needs prior training nor consumes time in comparing signs with those present in the database; rather, signs captured from live video are dynamically recognized.

THE PROPOSED METHOD

In the proposed method, all digits [0-9] and alphabet signs having thumb visible are recognized following the algorithm of seven phases. In the first phase, image tightly bounding the hand posture is extracted for which edges are detected in second phase. Third phase is to search out thumb position through Template Matching. Fingers are distinguished from one another in fourth phase. Groups are identified in fifth phase using positions and status of the distinguished fingers. Signs are recognized in sixth phase while the last phase is to produce output, both in audio and visual forms. The abstract level diagram of the algorithm is given in Figure 1.

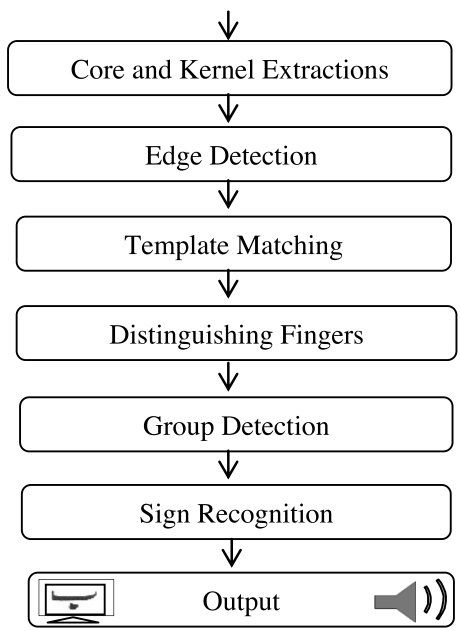


Fig. 1. Schematic of the proposed System

Core and kernel extractions

Core image, in the proposed algorithm, is the portion of captured scene most likely to hold hand posture while Kernel is the image that encloses exactly the hand gesture. The very first phase is to segment out the portion containing hand posture from the rest of the whole input scene. First of all, Core image is extracted from the whole captured scene and then the Kernel image. The process to obtain the meaningful hand posture (Kernel) is shown in Figure 2.

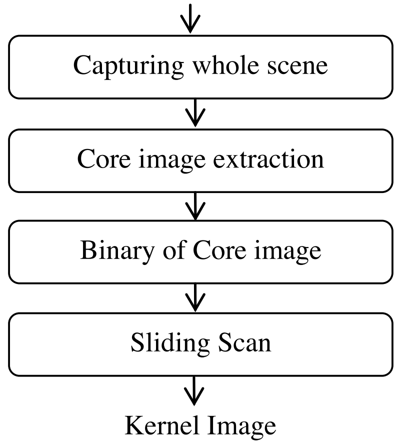
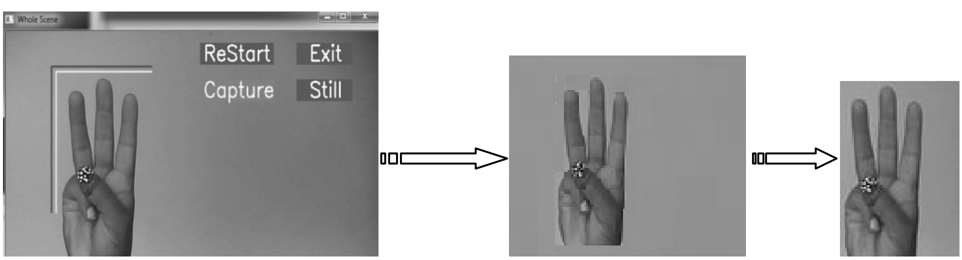


Fig. 2. Schematic of Kernel extraction

The step-wise process of extracting the image sign is shown in Figure 3.



(a) (b) (c)

Fig. 3. (a) Whole scene (b) the captured frame (c) Extracted image from ROI

Core image is a 265×245 pixels image extracted in RGB from the dedicated region of running video at real time as shown in Figure 4.

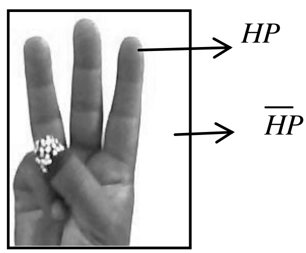


Fig. 4. The extracted Core image

If *Cr*, *Kr* and *HP* represents Core image, Kernel image and the Hand Posture respectively then,

$$\mathrm{Cr} = \mathrm{HP} \cup \overline{\mathrm{HP}} \tag{1}$$

The white-spaces of *Cr* containing mostly the meaningless background are removed to achieve faster computation. Kernel (*Kr*) is the image tightly bounding the hand posture. To get *Kr*, *Cr* is first assigned to a temporary image (T_{img}) which is converted to binary for the purpose of finding out the T_m (Topmost), L_m (Leftmost) and R_m (Rightmost) pixels enclosing the HP as shown in Figure 5. Considering the Rows and Columns of *Cr* enclosed inside T_m, L_m and R_m, *Kr* is obtained from the *Cr*.

$$Kr = HP \tag{2}$$

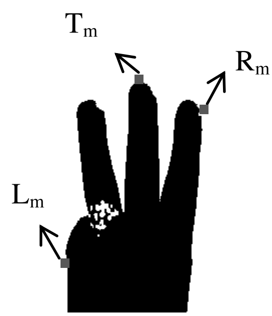


Fig. 5. Binary of *Cr* with L_m, T_m and R_m pixels

Edge detection

Before the detection of edges, slice of the scanned RGB image confined by T_m, L_m and R_m is assigned to *Kr*, as shown in Figure 6(a). *Kr* with rows *m* and columns *n* extracted from rows *r* and columns *c* of *Cr*, exactly circumscribes the hand posture.

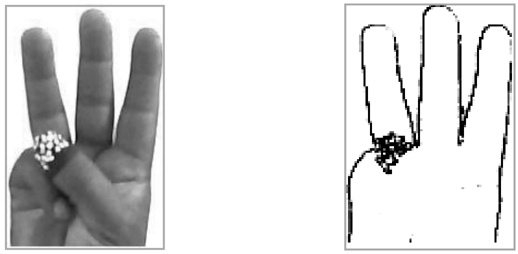
$$m = \bigcup_{r=Tm}^{C_r.rows} (r) \tag{3}$$

$$n = \bigcup_{c=Lm}^{Rm} (c) \tag{4}$$

Edges are detected using Sobel operator for efficient results. For each point of *Kr* the gradient is obtained as *G;*

$$\mathrm{G} = \sqrt{\left(\frac{\partial \mathrm{Kr}}{\partial \mathrm{x}}\right)^2 + \left(\frac{\partial \mathrm{Kr}}{\partial \mathrm{y}}\right)^2} \tag{5}$$

The edges detected Kernel is shown in Figure 6(b).



(a) (b)

Fig. 6. (a) The RGB of kernel (b) edge detected kernel

Template matching

For easy identification of thumb, a unique Template-sign is designed over thumb jacket (covering) that can be seen in all signs' figures. Thumb's template (T_t) is searched out in *G* of *Kr* using the *Square Difference Matching* method.

$$\mathrm{Kr} = \sum_{(L_m, T_m)} [\, (L_m, T_m) - \mathrm{Kr}(T_x + L_m\,,\ T_y + T_m)]^2 \tag{6}$$

Where T_x and T_y represent width and height of the template image respectively.

Distinguishing fingers

Excluding thumb, the rest of the fingers are first detected and then distinguished from each other. The algorithm follows the *Sliding* technique of scanning to detect finger's tip. So, *Kr* is scanned from top-left to bottom-right for the first black pixel to encounter. To make the system signer-independent and to avoid the distance problem, *Finger's Width* (Fw) is calculated once for the top-most finger. This is achieved by going five pixels deep from the finger's tip and calculating the Euclidean distance (Ed) between the left $Pl(xl, yl)$ and right $Pr(xr, yr)$ edge-pixels of the finger as shown in Figure 7(b).

$$\mathrm{Ed} = \sqrt{(\mathrm{xr} - \mathrm{xl})^2 + (\mathrm{yr} - \mathrm{yl})^2} \tag{7}$$

To avoid the slight width differences among fingers, *Fw* is increased by the addition of five pixels which is then assumed constant for all the four fingers.

$$\mathrm{Fw} = \mathrm{Ed} + 5 \tag{8}$$

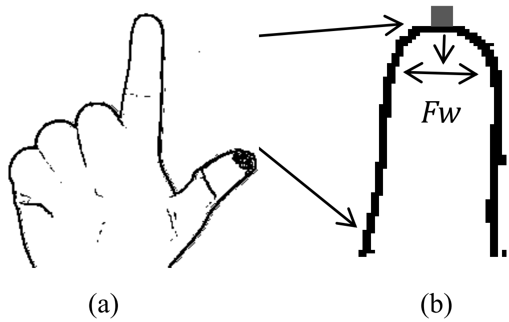


Fig. 7. (a) Finger's tip (b) finger's width

The same method of sliding is repeated to find out topmost pixel $Tp_{(x,y)}$ for the detection of the remaining three fingers. To avoid the scanning of an already identified finger, $Tp_{(x,y)}$ lying inside *Fw* of any of the previously identified fingers are omitted.

$$Tp_{(x,y)} \notin \{ Fw_i \}$$

Here 'i' representing the four fingers, ranges from 1 to 4.

The next step is to distinguish each finger from the rest. For this purpose, leftmost $Tp_{(x,y)}$ is found out from the set of known $Tp_{i(x,y)}$. The finger thus accessed first from left is sure to be SMALL finger, the next will be RING finger and so on as shown in Figure 8.

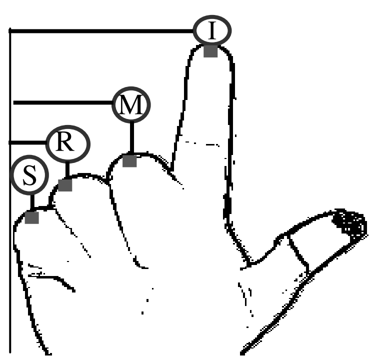


Fig. 8. Small (S), Ring (R), Middle (M) and Index (I) fingers recognized

Group detection

The alphabet signs having thumb visible are grouped into G1, G2, G3 while the ten digits (0 to 9) signs are categorized into groups namely G4, G5, G6 and G7 given as,

$$
S = \begin{cases}
G1 & \text{if} \quad Thumb.x\text{-}20 \leq Index.x \\
 & \quad\quad AND\ Thumb.x > Middle.x \\
G2 & \text{if} \quad Thumb.x < Index.x \\
G3 & \text{if} \quad Thumb. \geq Index.x{+}20 \\
G4 & \text{if} \quad Thumb.x - 10 \leq Index.x \\
G5 & \text{if} \quad Thumb.x > Totalcolumns\ \text{-}10 \\
 & \quad\quad AND\ Thumb.y < Mid\text{-}row \\
 & \quad\quad AND\ Closed\text{-}h \\
G6 & \text{if} \quad Thumb.x > Totalcolumns\ \text{-}10 \\
 & \quad\quad AND\ Thumb.y < Mid\text{-}row \\
G7 & \text{if} \quad Thumb.x > Totalcolumns\ \text{-}10 \\
 & \quad\quad AND\ Thumb.y > Mid\text{-}row
\end{cases}
$$

Sign recognition

The distinguished fingers with their respective positions are fed to the engine; especially designed for *isolated* signs of PSL. The system dynamically assigns x and y positions of each finger to a *Point* variable declared with the name of that finger. The Engine checks the respective positions of distinguished fingers and thus recognizes the signs.

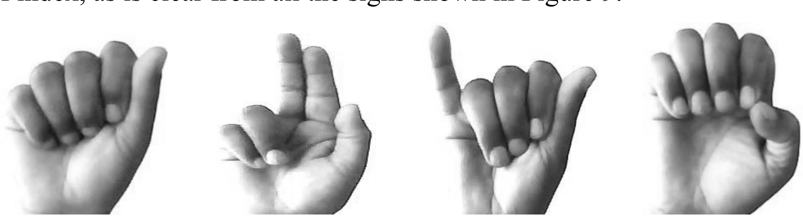
Standard deviation (SD) about y-axis of all the fingers, except *thumb*, is calculated with the following formula to find out whether the fingers are down are not.

$$\sigma = \sqrt{\frac{1}{4}\sum_{i-1}^{4}(y_i - \mu)^2} \tag{9}$$

If all the four fingers are down, σ will be less than 6 while in the signs where some fingers are down and others are raised the value of σ will be greater than 6.

G1 sign recognition

In *G1* signs, thumb being adjacent to index finger rests within a limit of 20 pixels at right of index, as is clear from all the signs shown in Figure 9.



(a) (b) (c) (d)

Fig. 9. Signs of G1 (a) ALIF (b) JAA (c) CHOTE YAA and (d) WAWO

Signs of *G1* are recognized using the following decision control statements.

$$G1 = \begin{cases} ALEF & \text{if} \quad ((SD<6) \\ & \quad AND\ (\ Thumb.y<Index.y)) \\ JAA & \text{if} \quad ((SD>6) \\ & \quad AND\ (\ Smallf.y>Ring.y\ ) \\ & \quad AND\ (Middle.y<Index.y)) \\ WAWO & \text{if} \quad (\ (SD<6) \\ & \quad AND\ (Thumb.y>Index.y+40)) \\ CHOTE\ \ YAA & \text{if} \quad ((SD>6) \\ & \quad AND\ (Thumb.y<Index.y) \\ & \quad AND\ (Small.y<Index.y) \\ & \quad AND\ (Index.y<=Middle.y)) \end{cases}$$

G2 sign recognition

In *G2* signs, thumb lies at left of index finger as in shown in Figure 10.

(a) (b) (c) (d) (e)

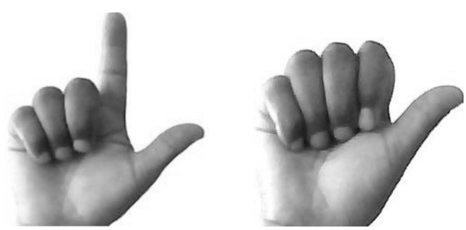
Fig. 10. *G2* Signs (a) BAA (b) KAF (b) YAA (c) ZOWAD and (d) SEEN

The logic structure of *G2* Signs is as,

$$G2 = \begin{cases} BAA & if & (SD<8)AND(Middle.y<Ring.y) \\ & & AND(Ring.y<Index.y)AND(Index.y<Small.y) \\ ZOWAD & if & (SD>6)\ AND\ (Index.y<Small.y) \\ & & AND\ (Small.y<Ring.y) \\ & & AND\ (Abs(Ring.y-Middle.y)<5) \\ SEEN & if & (SD<6)\ AND\ (Thumb.y>Index.y+10) \\ KAAF & if & (Abs(Middle.y-Index.y)<5) \\ & & AND(Abs(Ring.y-Small.y)<5) \\ & & AND(Thumb.x<Index.x) \\ & & AND(Thumb.x>Middle.x) \\ YAA & if & (SD<6)\ AND\ (Thumb.y>Index.y+40) \end{cases}$$

G3 sign recognition

The signs in which thumb is at least 20 pixels away at right from index are grouped in *G3*, see Figure 11.



(a) (b)

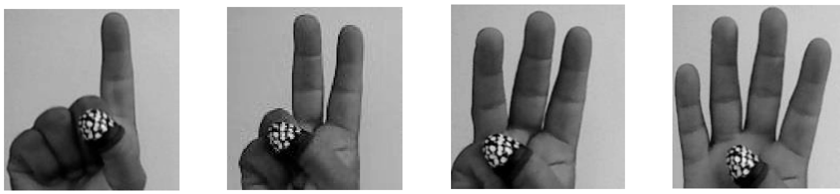
Fig. 11. *G3* Signs (a) LAAM (b) SOWAD

The two signs of *G3* are distinguished using the following rule.

$$G3 = \begin{cases} LAAM & if & ((Index.y < Middle.y) \\ & & AND(Index.y < Ring.y) \\ & & AND(Index.y < Small.x) \\ & & AND(Thumb.x > Index.x+20)) \\ SOWAD & if & ((Thumb.x > Index.x+20) \\ & & AND\ (SD < 6)) \end{cases}$$

G4 sign recognition

G4 contains signs of digits from 1 to 4, as shown in Figure 12.

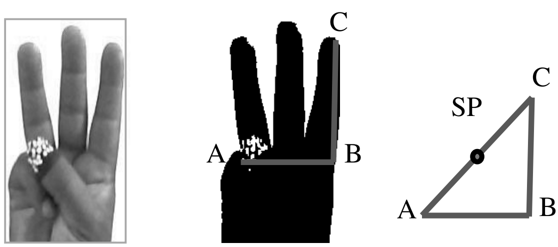


(a) (b) (c) (d)

Fig. 12. G4 signs for (a) one (b) two (c) three and (d) four

To recognize G4 signs, first raised status of index finger is checked by counting sequence of black pixels in the upper right part, all adjacent to right-most column of *Kr* as shown in Figure 13(b).The length of index and width of hand are treated as perpendicular and base respectively. Scanning Point (SP) is the mid-point of hypotenuse '*AC*' which is computed using Pythagorean Theorem as,

$$AC = \sqrt{AB^2 + BC^2} \tag{10}$$



(a) (b) (c)

Fig. 13. (a) *Kr* of sign for 3 (b) Binary of Kernel with base AB and perpendicular BC (c) the triangle ABC with *SP*

Horizontal Scanning Row (HSR); the row passing through each of the raised fingers is selected containing all columns (x_i) at height 'y' of SP, as shown in Figure 14.

$$HSR = SP.x_i,\ SP.y \tag{11}$$

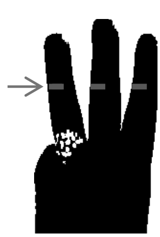


Fig. 14. Sign of three where three raised fingers detected

The entire HSR at height *SP.y* will be checked for raised fingers using the loop structure, given as,

while(Not End-of-HSR)

if(Black-Pixel)

increase Black-pixel-count by one

if(Black-pixel-count =5)

Count the Finger

if(While-Pixel)

increase White-pixel-count by one

if(White-pixel-count =5)

Reset Black-pixel-count to 0

G5 sign recognition

Signs for 0 and 8 resemble each other and are combined in G5. In both signs, a closed-hole is formed varying in areas as shown in Figure 15 and Figure 16.

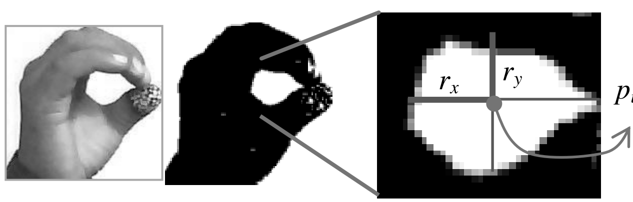


Fig. 15. Sign of zero with the closed-hole

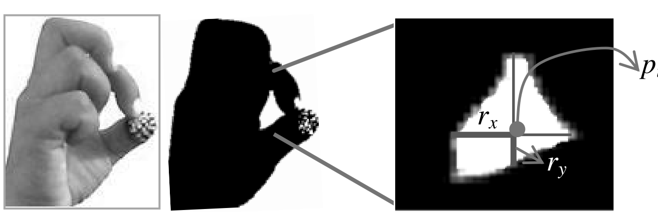


Fig. 16. Sign of eight with the closed-hole

The closed-hole formed in both the signs is treated as circle, the least of horizontal radius (r_x) and vertical radius (r_y) are calculated from the point of intersection (*Pi*). Area of the white region is supposed to be the sum of A1 and A2.

$$A1 = \pi(r_x)^2 \tag{12}$$

$$A2 = \pi(r_y)^2 \tag{13}$$

$$G5 = \begin{cases} 0 & \text{if} \quad ABS(A1 - A2) \leq 10 \\ & \quad AND\ Thumb.y \leq Tm+50 \\ 8 & \text{if} \quad ABS(A1 - A2) > 10 \\ & \quad AND\ Thumb.y > Tm+70 \end{cases}$$

G6 sign recognition

This group contains signs for 6 and 7. In the sign for six, the top-most pixel lies somewhere at the mid of *Kr*'s width with relaxation of five pixels both at left and right, as shown in Figure17.

$$\frac{Kr.n}{2} - 5 \leq Tm \leq \frac{Kr.n}{2} + 5 \tag{14}$$

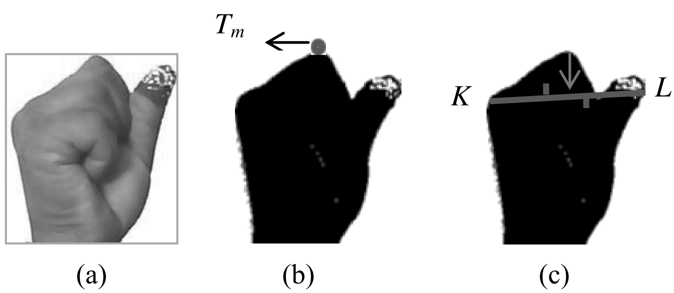


Fig. 17. (a) Sign of Six with (b) T_m Pixel and (c) width *KL* of Kernel

For the sign of 7, Euclidian distance between thumb's position 'P' and down-most black pixel 'Q', is calculated. The distance is supposed to be equal to the width of Kr with relaxation of ten-pixels, as shown in Figure 18.

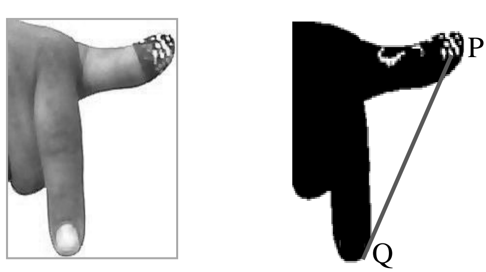


Fig. 18. Sign of seven with positions of thumb (P) and Down-most black pixel (Q)

$$\text{PQ} = \sqrt{(Px - Qx)^2 + (Py - Qy)^2} \tag{15}$$

$$G6 = \begin{cases} 6 & \text{if} \quad ABS(Kr.Columns - Tm) \leq 5 \\ & \quad\quad AND\ Thumb.y - Tm \leq 20 \\ 7 & \text{if} \quad ABS(Kr.Columns - PQ) \leq 10 \\ & \quad\quad AND\ Thumb.y \leq 10 \end{cases}$$

G7 sign recognition

Signs for 5 and 9 are grouped in G7. In the sign for 9, origin $O_{(x,y)}$ is obtained from T_m and R_m of *Kr* as shown in Figure 19.

$$O_{(x,y)} = \left(Tm_{(row)}\ ,\ Rm_{(column)}\right) \tag{16}$$

$$r_1 = \sqrt{(O_x - Rm)^2} \tag{17}$$

Thumb finger must lie outside the inner arc and inside the outer arc, where $r_2 = 2r_1$.

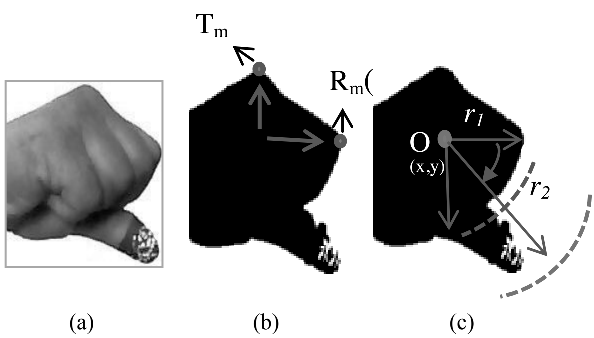


Fig. 19. (a) Sign of nine with (b) topmost and rightmost (c) inner and outer arcs

For the sign for 5, all fingers are raised while thumb lies at lower right, as shown in Figure 20.

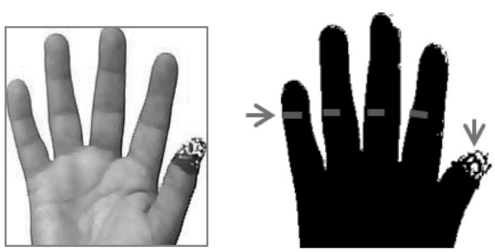


Fig. 20. The sign for five

$$
\mathrm{G4} = \begin{cases} 9 \quad \text{if} \quad Thumb_y > O_y + r_1 * \sin(\theta) & \forall\, \theta \in \{30, \ldots 90\} \\ \qquad\quad AND\ Thumb_y < O_y + r_2 * \sin(\theta) & \forall\, \theta \in \{30, \ldots 90\} \\ 5 \quad \text{if} \quad Thumb_x > Kr.Columns - 20 & \\ \qquad\quad AND\ Thumb_y > P.y\ AND\ RaisedFingers{=}4 & \end{cases}
$$

Output

The system engine, after recognizing the exact sign, invokes the last module dealing with both audio and visual outputs. The detected sign is pronounced in the exact standard Urdu accent and is displayed in a 200×200 pixels image form.

EXPERIMENTAL RESULT AND ANALYSIS

The model was implemented in Microsoft Visual Studio 2010 with the library of OpenCV. A Core i5 Laptop running at 2.5GHz was used for the development and testing of the system. Each captured frame held four buttons which were highlighted with mouse move-over, as shown in Figure 21. A new frame was captured after providing a healthy time of 30ms for the signer to pose. One can directly capture sign by clicking the Capture button if hand gesture is posed before the expiry of 30ms. Restart button is to reset the system for a new sign. Clicking over the Still button halts the whole system while Exit button is to quit. To test the system, twenty trials were performed for each sign by five signers.

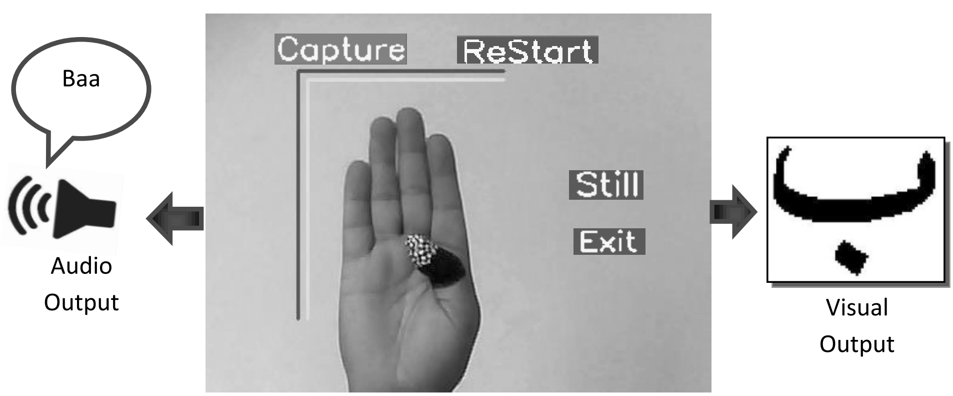
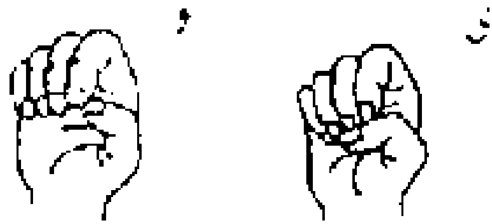


Fig. 21. Captured image with specified ROI

Details of all signs, correct, false and missed detections, are shown in Table 1. The following formula was used for accuracy calculation yielding an accuracy of 84.2%.

Accuracy (in %age) = *(No. of Correct Recognition –No of Failed Recognition)/ Total Signs * 100*

The noteworthy fact about the method is its accurate recognition of very similar signs like that of WAWO and YAA shown in Figure 22 (a) and Figure 22 (b) respectively.



(a) (b)

Fig. 22. The closely resembled signs of (a) WAWO and (b) YAA

Table 1. Statistics of correct, false and missed recognition of the signs

Digits/Alphabets	Correct detection	False detection	Missed detection	TOTAL
0	17	2	1	20
1	20	0	0	20
2	20	0	0	20
3	18	2	0	20
4	18	2	0	20
5	20	0	0	20
6	20	0	0	20
7	19	1	0	20
8	17	3	0	20
9	15	3	2	20
ALEF	18	2	1	20
JAA	18	1	1	20
WAWO	17	1	2	20
CHOTE YA	18	2	1	20
BAA	20	0	0	20
KAF	19	1	0	20
ZOWAD	19	1	0	20
SEEN	20	0	0	20
YAA	18	0	2	20
LAAM	19	0	1	20
SOWAD	18	1	1	20
Total	**388**	**22**	**12**	**420**

Taking accuracy, light-sensitivity, cost and signer dependency as frontiers of analysis; as illustrated in Table 2, the designed system is relatively efficient as compared with known systems.

Table 2. Analysis of some of the state of the art signs recognition systems

Authors	Year	Acquisition system	Classification	Accuracy	Light Sensitivity	Cost	Signer dependency
Alvi *et al.*	2005	Data Gloves	Template Matching	69%	No	High	No
Sumaira *et al.*	2008	Colour Glove	Fuzzy Classifier	94%	Yes	Low	Yes
Nagarajan *et al.*	2012	Skin Colour Segmentation	Convexity defects	96%	Yes	Low	Yes
Alia *et al.*	2011	Skin Colour Segmentation	HMM	82%	Yes	Low	Yes
Jerde *et al.*	2003	CyberGlove with sensors	Discriminate Analysis	95%	No	High	No
Holden *et al.*	2001	Colour coded Gloves	HMM	95%	Yes	Low	Yes
Ong *et al.*	2009	Kinect Tracking	SP Trees	55%	No	High	No
Raees *et al.*	2014	Thumb template	Pixels analysis	84%	No	Low	No

CONCLUSION AND FUTURE WORK

The system is specifically designed for the recognition signs from PSL but its applicability and accuracy testify that the algorithm may be pursued for any sign language, especially for Arabic Sign Language. Orientation of hand exceeding ten degrees in either axis is the main reason for false detection. The 16% failure of the system is done to tilt or orientation of hand during gesturing. We have aimed to enhance the algorithm in future so that orientation up to large extent could be tolerated.

REFERENCES

Al-Jarrah, O. & Alaa, H. 2001. Recognition of gestures in Arabic sign language using neuro-fuzzy systems. Artificial Intelligence, **133**(1-2): 117-138.

Alia, O. M., Mandava R., Ramachandram D. & Aziz M. E., 2009. A novel image segmentation algorithm based on harmony fuzzy search algorithm. In international conference of soft computing and pattern recognition, pp 335–340.

Alvi, A. K., Azhar, M. Y. B., Usman, M., Mumtaz, S., Rafiq, S., Rehman, R. U. & Ahmed, I. 2005. Pakistan sign language recognition using statistical template matching. International Conference on Information Technology, (ICIT 2004), Istanbul, Turkey pp. 108-111.

Assaleh K. & Al-Rousan, M. 2005. Recognition of Arabic sign language alphabet using polynomial classifiers. EURASIP Journal on Applied Signal Processing, **13**, pp. 2136-2146.

Chalechale A., & Safae, F. 2008. Visual-based interface using hand gesture recognition and object tracking. Iranian Journal of Science and Technology; Transaction B, Engineering Shiraz University, **32**: 279–293.

Doliotis, P., Mcmurrough, C., Eckhard, D. & Athitsos, V. 2011. Comparing gesture recognition accuracy using colours and depth information. Conference on Pervasive Technologies Related to Assistive Environments, Heraklion, Crete, Greece.

Erdem, U. M. & Sclaroff, S. 2002. Automatic detection of relevant head gestures in American Sign Language communication. International Conference on Pattern Recognition,Vol.1, BRAC University Dhaka, Bangladesh.

Fan, Z. & Ling S. 2014. Weakly-supervised cross-domain dictionary learning for visual recognition, International Journal of Computer Vision, **109**(1-2): 42-59.

Fang, G., Wen, G. & Zhao, D. 2007. Large-vocabulary continuous sign language recognition based on transition-movement models. IEEE Transactions on Systems, Man & Cybernetics, Part A: Systems and Humans, **37**(1): 305-314.

Holden, E. & Owens., R. 2001. Visual sign language recognition. Multi-Image Analysis, Lecture Notes in Computer Science, Springer, pp. 270–287.

Jerde, T. E., Soechting, J. E. & Flanders, M. 2003. Biological constraints simplify the recognition of hand shapes. Transactions on Biomedical Engineering, **50**(2): 265–269.

Kadous M. W. 1996. Machine recognition of Auslan signs using power gloves towards large-lexicon recognition of sign language. Proceedings of the Workshop on the Integration of Gestures in Language & Speech, pp. 165-174, Delaware and Wilmington, Delaware .

Kuch, J. J. & Huang, T. S. 1995. Vision-based hand modelling and tracking for virtual teleconferencing and tele-collaboration. IEEE International Conference on Computer Vision, Cambridge, MA, USA.

Ling, S., Li, L., & Xuelong, L. 2014. Feature Learning for Image Classification via Multi objective Genetic Programming, IEEE Transactions on Neural Networks and Learning Systems, **25**(7): 1359-1371.

Liu, X. & Fujimura, K. 2004. Hand gesture recognition using depth data. In: Proceedings of FGR, Seoul, Korea.

Muir, L. & Leaper, S. 2003. Gaze tracking and its application to video coding for sign language. Picture Coding Symposium, the Robert Gordon University, School hill, Aberdeen, UK.

Nagarajan, S., Subashini, T. & Ramalingam, V. 2012. Vision Based Real Time Finger Counter for Hand Gesture Recognition. International Journal of Technology, **2**(2): 1-5, CPMR-IJT.

Ong, E. J., Cooper, H., Pugeault, N. & Bowden, R. 2012. Sign language recognition using sequential pattern trees. In International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2200–2207.

Raees, M., Sehat U. 2014. Alphabet signs recognition using pixels-based analysis. 5th Conference on Language and Technology (CLT14), DHA Suffa University (DSU), Karachi.

Rajeshree, R. & Manesh, K. 2009. Gesture recognition by thinning method. International Conference on Digital Image Processing, IEEE Computer Society, Bangkok, pp. 284-287.

Ravikiran, J., Kavi, M., Suhas, M., Dheeraj, R., Sudheender, S. & Nitin, V. P. 2009. Finger detection for sign language recognition. Proceedings of the International Multi Conference of Engineers and Computer Scientists, Hong Kong.

Saad, A., Jonas, B. & Hedvig, K. 2012. Visual recognition of isolated Swedish sign language signs. PhD Thesis Cornell University Ithaca, New York, United States.

Shanableh, T. 2007. Arabic sign language recognition in user independent mode. IEEE International Conference on Intelligent and Advanced Systems, Kuala Lumpur, Malaysia.

Simon, L., Marco, D. & Block, B. 2011. Sign language recognition with kinect. MS Thesis, Freie Universität Berlin, Germany.

Starner, T., Weaver, J. & Pent I. A. 1998. Real-time American sign language recognition using desk and wearable computer based video. IEEE Transactions on Pattern Analysis & Machine Intelligence, **20**(2): 1371-1375.

Subha, R. & Balakrishnan, G. 2011. Real-time Indian sign language recognition system to aid deaf-dumb people. IEEE 13th International Conference on Communication Technology. Indian Institute of Technology Kharagpur, Kharagpur, India.

Sumera, K., Younus, M. & Shaleeza, S. 2008. Recognition of gestures in Pakistani sign language using fuzzy classifier. 8th International Conference on Signal Processing, Computational Geometry and Artificial Vision, (ISCGAV'08), Rhodes, Greece.

Vezhnevets, S. V. & Reeva, A. 2003. A Survey on pixel-based skin colour detection techniques. Proceedings of Graphicon, Moscow, Russia.

Von, A., Knorr, M., Kraiss, K. & Jonghwa, K. 2008. The significance of facial features for automatic sign language recognition. 8th IEEE International Conference on Automatic Face Gesture Recognition, Amsterdam, Netherlands.

Wen, G., Jiyong, M., Jangqin, W. & Chunli, W. 2000. Sign language recognition based on HMM/ANN/DP. International Journal of Pattern Recognition and Artificial Intelligence, **14**(5): 587-602.

Yang H. D. & Lee S. W. 2010. Robust sign language recognition with hierarchical conditional random fields. In 20th International Conference on Pattern Recognition, Istanbul Turkey.

Zabulis, X., Baltzakisy, H. & Argyroszy, A. 2007. Vision-based hand & gesture recognition for human-computer interaction. Institute of Computer Science Foundation for Research and Technology, University of Crete Heraklion, Crete, Hellas, Heraklion.

Zhang, L., Zhen, X., & Shao, L. 2014. Learning object-to-class kernels for scene classification. IEEE Transactions on Image Processing, **23**(8): 3241-3253.

Submitted: 9/2/2015

Revised: 7/7/2015

Accepted: 9/7/2015