Learn2Sign: Explainable AI for Sign Language Learning

Prajwal Paudyal, Junghyo Lee, Azamat Kamzin, Mohamad Soudki, Ayan Banerjee, Sandeep K.S. Gupta
Arizona State University
Tempe, Arizona, United States
(ppaudyal,jlee375,akamzin,msoudki,abanerj3,sandeep.gupta)@asu.edu

ABSTRACT

Languages are best learned in immersive environments with rich feedback. This is specially true for signed languages due to their visual and poly-componential nature. Computer Aided Language Learning (CALL) solutions successfully incorporate feedback for spoken languages, but no such solution exists for signed languages. Current Sign Language Recognition (SLR) systems are not interpretable and hence not applicable to provide feedback to learners. In this work, we propose a modular and explainable machine learning system that is able to provide fine-grained feedback on location, movement and hand-shape to learners of ASL. In addition, we also propose a waterfall architecture for combining the sub-modules to prevent cognitive overload for learners and to reduce computation time for feedback. The system has an overall test accuracy of 87.9 % on real-world data consisting of 25 signs with 3 repetitions each from 100 learners.

CCS CONCEPTS

• **Human-centered computing** → **Interaction design**; • **Computing methodologies** → **Artificial intelligence**; • **Applied computing** → **Interactive learning environments**.

KEYWORDS

Explainable AI; Sign Language Learning; Computer-aided learning

ACM Reference Format:
Prajwal Paudyal, Junghyo Lee, Azamat Kamzin, Mohamad Soudki, Ayan Banerjee, Sandeep K.S. Gupta. 2019. Learn2Sign: Explainable AI for Sign Language Learning. In *Joint Proceedings of the ACM IUI 2019 Workshops, Los Angeles, USA, March 20, 2019* , 7 pages.

1 INTRODUCTION

Signed languages are natural mediums of communication for the estimated 466 million deaf or hard of hearing people worldwide [16]. Families and friends of the deaf can also benefit from being able to sign. The Modern Language Association [2] reports that the enrollment in American Sign Language (ASL) courses in the U.S. has increased nearly 6,000 percent since 1990 which shows that interest to acquire sign languages is increasing. However, the lack of resources for self-paced learning makes it difficult to acquire, specially outside of the traditional classroom setting [26].

The ideal environment for language learning is immersion with rich feedback [27] and this is specially true for sign languages [9]. Extended studies have shown that providing item-based feedback in CALL systems is very important [35]. Towards this goal, extensive language learning softwares for spoken languages such as Rosetta Stone or Duolingo support some form of assessments and automatic feedback [31]. Although, there are numerous instructive books [23], video tutorials or smartphone applications for learning popular sign languages, there hasn't yet been any work towards providing automatic feedback as seen in Table 1. We conducted a survey [13] of 52 first-time ASL users (29M, 21F) in 2018 and 96.2 % said that reasonable feedback is important but lacking in solutions for sign language learning (Table 2).

Table 1: Some ASL learning applications for smartphones.

Application	Can Increase Vocab	Feedback
ASL Coach	No	None
The ASL App	No	None
ASL Fingerspelling	No	None
Marlee Signs	Yes	None
SL for Beginners	No	None
WeSign	No	None

Studies show that elaborated feedback such as providing meaningful explanations and examples produce larger effect on learning outcomes than just feedback regarding the correctness [35]. The simplest feedback that can be given to a learner is whether their execution of a particular sign was correct. State-of-the-art SLR and activity recognition systems can be easily trained to accomplish this. However, to truly help a learner identify mistakes and learn from them, the feedback and explanations generated must be more fine-grained.

The various ways in which a signer can make mistakes during the execution of a sign can be directly linked to how minimum pairs are formed in the phonetics of that language. The work of Stokoe postulates that the manual portion of an ASL sign is composed of 1) location, 2) movement and 3) hand-shape and orientation [30]. A black box recognition system cannot provide this level of feedback, thus there is the need for an explainable AI system because feedback from the system is analogous to explanations for its final decision. Non-manual markers such as facial expressions and body gaits also change the meaning of signs to some extent but they are less important for beginner level language acquisition, so these will be considered for future work.

Studies have also shown that the effect of feedback is highest if provided immediately [35], thus feedback systems should be real-time. The requirement for immediate feedback also restricts the usage of complicated learning algorithms that require heavy

IUI Workshops'19, March 20, 2019, Los Angeles, USA

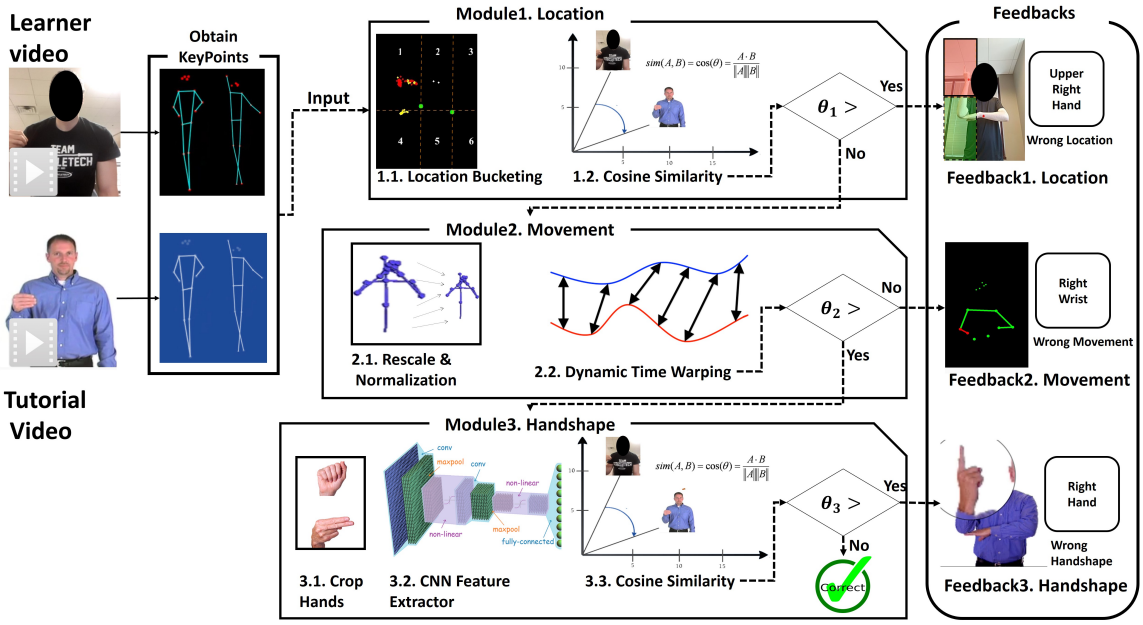


Figure 1: System Model for Feedback.

computing [6] and extensive training. The usability and usefulness of applications is enhanced if learning is self-paced, learners are allowed to use their own devices, and the learning vocabulary can be easily extended. However, current solutions for SLR require re-training to support unseen words and large datasets initially. To solve these challenges, we designed Learn2Sign(L2S), a smartphone application that utilizes explainable AI to provide fine-grained feedback on location, movement, orientation and hand-shape for ASL learners. L2S is built using a waterfall combination of three non-parametric models as seen in Figure1 to ensure extendibility to new vocabulary. Learners can use L2S with any smartphone or computer with a front-facing camera. L2S utilizes a bone localization technique proposed by [17] for movement and location based feedback and a light-weight pre-trained Convolutional Neural Network (CNN) as a feature extractor for hand-shape feedback.

The methodology and evaluations are provided in Sections 3 and 4. As part of the work, we collected video data from 100 users executing 25 ASL signs three times each. The videos were recorded by L2S users in real-world settings without restrictions on device-type, lighting conditions, distance to the camera or recording pose (sitting or standing up). This was to ensure generalization to real-world conditions, however, this makes the dataset more challenging. More details about the resulting dataset of about 7500 instances can be found in [13].

2 RELATED WORK

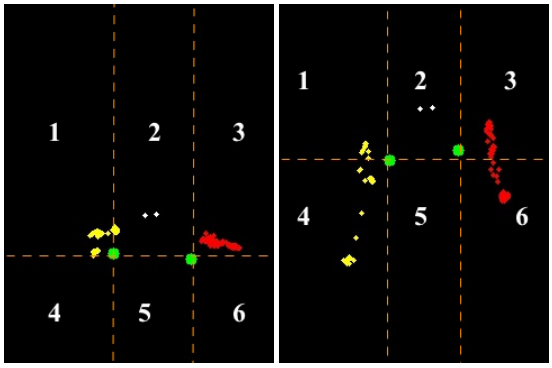
There have been many works on providing meaningful feedback for spoken language learners [8, 21, 22]. On the practical side, Rosetta Stone provides both waveform and spectrograph feedback for pronunciation mistakes by comparing acoustic waves of a learner to that of a native speaker [31]. There has also been some recent work

Table 2: Survey Results from 52 Users of the Application.

Category	Response	
Importance of Feedback	Yes: 96.2%	No: 3.8%
Movement Feedback	Correctness: 9.6% Colored Bones: 1.9% Sentence: 15.4% Correctness+Sentence: 9.6% All: 65.3	
Handshape Feedback	Circle around handshape: 63.5% Actual handshape: 36.5%	
Self-Assessment	Not helpful: 1.9% Somewhat: 5.8% Very helpful: 93.3%	
Expandability	Not helpful: 5.8% Somewhat: 7.7% Very helpful: 86.5%	

on design principles for using Automatic Speech Recognition (ASR) techniques to provide feedback for language learners [36]. Sign Language Recognition (SLR) is a research field that closely mirrors ASR and can potentially be utilized by systems for sign language learning. However, to the best of our knowledge, no such system exists. This can be explained by the inherent difficulties in SLR as well as the lack of detailed studies on design principles for such systems. In this work, we propose some design principles and an explainable smart system to meet this goal.

Continuously translating a video recording of a signed language to a spoken language is a very challenging problem and has been



(a) TIGER mostly in bucket 1 for left-hand. **(b) DECIDE in buckets 3 and 6 for right-hand.**

Figure 2: Automatic bucketing for Location Identification for varying distances from camera. Left Wrist-Yellow, Right Wrist-Red, Eyes - White, Shoulders-Green.

tackled recently by various researchers with some success [6]. For the purposes of this application, such complex measures are not desirable, as they mandate extensive datasets for training and large models for translation which decreases their usability. Isolated Sign Language Recognition has the goal of classifying various sign tokens into classes that represent some spoken language words [11, 12, 18, 29]. Some researchers have utilized videos [14] while some others have attempted to use wearable sensors [18, 19] with varying performances. In this work, we utilize the insights and advances from such systems to help a new learner acquire the sign language words. To our knowledge, this work is the first attempt at such a practical and much needed application.

For this work, we require an estimation of human pose, specifically the estimates on the location of various joints throughout a video, known as keypoints. There have been several works towards this goal [3–5, 25, 32, 33]. Some of these works first detect the keypoints in 2D and then attempt to 'lift' that set to 3D space while others return the 2D coordinates of the various keypoints relative to the image. In order to fulfill the requirement to use pervasive cameras, we did not focus on the approaches that utilize depth information such as Microsoft Kinect [20]. Thus, we utilized the pose estimates from a Tensorflow JS implementation of a model proposed by Papandreou et al. [17] which can run on devices with or without GPUs (Graphical Processing Units).

3 METHODOLOGY

Stokoe proposed that a sign in ASL consists of three parts which combine simultaneously: the tab (location of the sign), the dez (hand-shape) and the sig (movement) [30]. Signs like 'HEADACHE' and 'STOMACH ACHE' that are similar in hand-shape and movement may differ only by the signing location. Similarly, there will be other minimal pairs of signs that differ only by the movement or hand-shape. Following this understanding, L2S is composed of three corresponding recognition and feedback modules.

3.1 User Interface

For initial data collection and for testing the UI, we developed an android application called L2S. We preloaded the application with 25 tutorial videos from Signing Savvy corresponding to 25 ASL signs [24]. The application has three main components: a) Learning Module b) Practice Module, and c) Extension.

3.1.1 Learning Module. The learning module of the L2S application is where all the tutorial videos are accessible. A learner selects an ASL word/phrase to learn and can then view the tutorial videos. The learner can pause, play, and repeat the tutorials as many times as needed. In this module, the learner can also record executions of their signs for self-assessment.

3.1.2 Practice Module. The practice module is designed to give automatic feedback to the learners. A learner selects a sign to practice and sets up their device to record their execution. After this, L2S determines if the learner performed the sign correctly. The result is correct if the sign meets the thresholds for movement, location, and hand-shape and a 'correct' feedback is given. If, the system determines that the learner did not execute the sign correctly, an appropriate feedback is provided as seen in Figure 1. Details about the recognition and feedback mechanisms is discussed in Section 3.4.

3.1.3 Extension Module. To extend the supported vocabulary of L2S, a learner can upload one or more tutorial videos from a source of their choosing. The application processes them for usability before they appear in the Learning Module as new tutorial sign(s).

3.2 Data Collection

We collected signing videos from 100 learners, for 25 ASL signs with three repetitions each in real-world settings using L2S app. Learners used their own devices, with no restrictions on lighting conditions, distance to the camera or recording pose (sitting or standing up). After reviewing a tutorial video, a learner was given a 5 s setup time before recording a 3 s video using a front-facing camera. Both the tutorial and the newly recorded video were then displayed in the same screen for the user to accept or reject. This self-assessment served not only as a review but it also helped prune incorrect data due to device or timing errors as suggested by the new learner survey in Table 2.

3.3 Preprocessing

Determining joint locations: Since, different devices record in different resolutions, all videos for learning, practice or extension are first converted to a 320*240 resolution. Then, PoseNet Javascript API for single pose estimation [17] was used to compute the estimated locations and confidence levels for the various keypoints as seen in Table 3. Figure 2 shows the estimated eyes, shoulder and wrist locations for the signs TIGER and DECIDE for all the frames in one video.

Normalization: There is a difference in scale of the bodies relative to the frame-size corresponding to the distance between the learner and the camera. This scaling factor can negatively impact recognition since the relative location, movement and hand-shape will vary with distance. We perform min-max normalization and zeroing based on the distance between the average estimated locations

Table 3: 4 out of 17 'keypoints' for one frame in a video

Part	Score	x	y
Left Shoulder	0.8325	180.0198	196.2646
Right Shoulder	0.78601	138.7879	195.5847
Left Wrist	0.6844	198.9818	223.4856
Right Wrist	0.1564	1.9084	211.0473

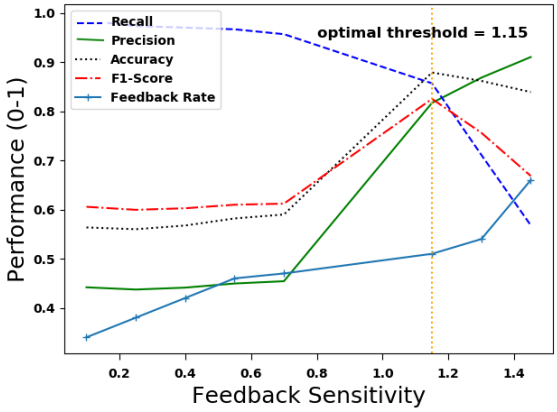


Figure 3: Feedback Sensitivity vs. Performance.

for the right and left shoulders throughout the video frame as suggested by [15]. Normalization was found to be specially important for correct movement recognition.

3.4 Recognition and Feedback

L2S is designed to give incremental feedback to learners for the various modalities in sign language: a) Location b) Movement and c) Hand-shape. The various models are arranged in a waterfall architecture as seen in Figure 1. If the location of signing was not correct, then immediate feedback is provided and the learner is prompted to try again. Similarly, if the movement of the elbows or the wrists for either hand was incorrect, the learner is prompted to try again. Finally, if the shape and orientation of either of the hands does not appear to be correct, a hand-shape based feedback is provided. Consequently, the learner can move on to a practice a new sign, only if all these modalities were sufficiently correct. A waterfall architecture was chosen in the final application over a linear weighted combination to make learning progressive and to decrease the cognitive load on the learner due to the potential of mistakes in multiple modalities. This architecture also helps to reduce the time taken for recognition and feedback since the models are stacked in an increasing order of execution time. Each of the feedback screens shown to the user also has a link to the tutorial video. Users can also manually tune the amount of feedback by altering the value of 'feedback sensitivity' in the application settings. Increasing this value alters the thresholds for each of the sub-modules so that the overall rate of feedback is increased. This involves a trade-off in performance which is summarized in Figure 3.

3.5 Location

To correctly and efficiently determine the location for signing, we first assume the shoulders stay fairly stationary throughout the execution of a sign. This is a fair assumption for ASL since there are no minimal pairs exclusively associated with a signer's shoulders. Then we divide the video canvas into 6 different sub-sections called *buckets* as seen in Figure 2. Then, as the learner executes any given sign, the location of both the wrist joints is tracked for each bucket resulting in a vector of length 6.

This same procedure is followed for the tutorials, and a cosine-based comparison between is done between the two vectors. A heuristic threshold that is determined during training is utilized as a cut-off point. If the resulting cosine similarity is lower than a threshold, some feedback is shown to the learner as seen in Figure 4. For each hand, the user's own video is replayed in Graphics Interchange Format (GIF) with a red highlight on the location section that was incorrect and a green highlight on the section of the frame where the sign should have been executed. A text feedback with details and a link to the tutorial is also provided and the learner is prompted to try again.

3.6 Movement

Determination of correct movement is perhaps the single most important feedback we can provide to a learner. We compute a segmental DTW distance between a learner and the tutorial using keypoints for the wrists, elbows and shoulders as suggested in [1]. Normalization as discussed in Section 3.3 was found to be very important. Experimental results showed that segmental DTW outperformed DTW or Global Alignment Kernel (GAK).

The dataset had a wide variation in the number of frames per video. It was found that this affected the distance scores adversely. Thus, as an additional step of preprocessing, the video with the higher number of frames was down-sampled before comparison and the segmental DTW is utilized to find the best sub-sample matching. Thresholds for the signs were determined experimentally using 10 training videos for each sign. If segmental DTW distance between a learner's recording and a tutorial was higher than the threshold for each arm section, then a movement-based feedback is provided as seen in Figure 1. A GIF is replayed to the user with the section(s) of the arm for which the movement was incorrect in red as seen in Figure 5b. A textual feedback is also generated with an explanation after which the user is prompted to watch the tutorial and try again.

3.7 Hand Shape and Orientation

ASL signs which are otherwise similar, may differ only by the shape or orientation of the hands. Since, CNNs have state-of-the-art image recognition results, we utilized Inception v3 or Mobilenet CNN depending on the device being used. A model that was pre-trained on ImageNet is retrained using hand-shape images from the training users. The wrist location obtained during pre-processing was used as a guide to auto-crop these hand-shape images. During recognition time, hand-shape images from each hand are extracted automatically in a similar way from a learner's recording. Then 6 images for each hand are passed separately through the CNN and the softmax layer is obtained and are concatenated together as seen

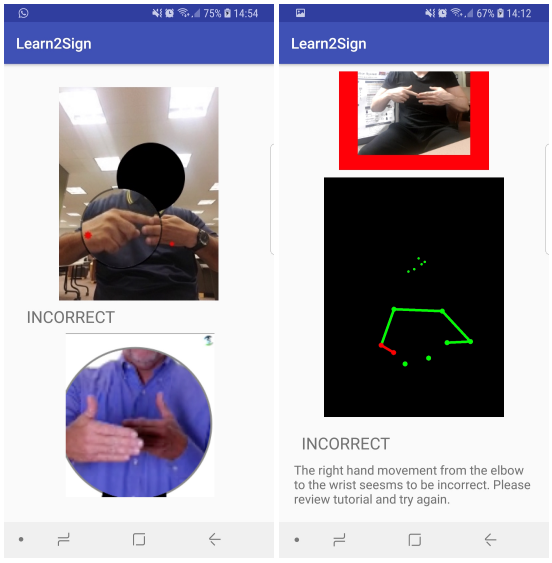
in Figure 1. Similar processing is done on the tutorial video to obtain a vector of the same length. Then a cosine similarity is calculated on the resultant vector. If the similarity between a learner's sign and that of a tutorial is above a set threshold for a sign, then the execution is determined to be correct, otherwise the hand-shape based feedback as seen in Figure 5 is provided.

Although the retrained CNN could theoretically be used as a classifier, we use it only as a feature extractor for cosine similarity to ensure that the system can extend to unseen classes. A new tutorial can then be effectively added to the system without the need for retraining. An analysis of the effectiveness of hand-shape and orientation recognizer is provided in Section 4.Similar to location and movement, feedback for hand shape and orientation is also provided in the form of a replay GIF and text. A zoomed in image of the incorrect hand shape is shown side by side with the correct image from a tutorial as seen in Figure 5(a).

4 RESULTS AND EVALUATION

An ideal system should give feedback to a learner only if their execution is incorrect. Giving unnecessary feedback for correct executions will hinder the learning process and decrease the usability. Conversely, providing sound and timely explanations for incorrect executions helps to improve utility and user trust. Smart systems such as L2S that use explainable machine learning tend to have a trade-off between explainability and performance which should be minimized.

The overall performance of the system was tested for 10 test users for a total of 750 signs. The training of the CNN for hand-shape feature extraction and optimal threshold determination was done using the remaining users. For each sign, 30 executions from the test dataset were taken as true class while 30 randomly selected executions from the pool of remaining signs was taken as



(a) Hand-shape feedback for AFTER. (b) Movement Feedback for ABOUT.

Figure 4: Feedback given by the app.

(a) HERE: Red box(upper): Detected Location, Green Box(lower): Correct Location.

(b) DEAF: Red box(upper): Detected Location, Green Box(lower): Correct Location.

Figure 5: Feedback for incorrect location for right hand.

incorrect class to avoid class imbalance. A pre-trained model from C3D [34] was retrained with the data we collected and was used as the baseline for comparison. This model has an accuracy of 82.3 % on UCF101 [28] and 87.7 % in YUPENN-Scene [7] datasets. The final recognition accuracy of C3D on L2S dataset using the same train-test split was 45.38 %. Our approach achieves a higher accuracy of 87.9 % while still offering explanations about its decisions in the form of learner feedback.

To obtain the results, data collected from one learner was selected at random and served as the *tutorial* dataset. Then each sign for each user in the test dataset was compared against the corresponding tutorial sign. The location module had an overall recall of 96.4 % and precision of 24.3 %. The lower precision is due to the fact that many signs in the test dataset had similar locations. We performed a test comparing only the signs 'LARGE' to the sign 'FATHER' and both the precision and recall were 100 %. The movement module had an overall recall of 93.2 % and a precision of 52.4 %. The hand-shape module had a recall of 89 % and a precision of 74 %. The overall model is constructed as a waterfall combination of all three models such that the movement model is executed only when the location was found to be correct, and the hand-shape model is executed only when both the location and movement were correct. The overall precision, recall, f-1 score and accuracies is summarized in Table 4.

5 DISCUSSION AND FUTURE WORK

We demonstrated the need for a feedback based technological solution for sign language learning and provided an implementation with a modular feedback mechanism. The user preference for the desired amount of feedback can be changed by altering the value for 'Feedback Sensitivity'. The trade-off between 'Feedback Sensitivity' and amount of feedback received as well as other performance metrics is summarized in Figure 3. Although, we designed our feedback mechanism based on principles from linguistics and user survey, only a large scale usage of such an application will provide definitive best practices for the most effective feedback. In such future studies, issues such as the extent of user control for determining

Table 4: Precision(P), Recall(R), F-1 Score (F1) and Accuracy(A) for 25 ASL tokens.

Sign	P	R	F1	A	Sign	P	R	F1	A	Sign	P	R	F1	A
About	0.92	0.71	0.80	0.85	**Decide**	0.91	0.55	0.69	0.74	**Here**	0.92	0.96	0.94	0.96
After	0.85	0.92	0.88	0.92	**Father**	0.86	0.86	0.86	0.92	**Hospital**	0.96	0.86	0.91	0.93
And	0.86	0.86	0.86	0.92	**Find**	0.54	0.81	0.65	0.81	**Hurt**	0.81	0.92	0.86	0.91
Can	0.96	0.77	0.85	0.89	**Gold**	0.88	0.81	0.84	0.89	**If**	0.79	0.90	0.84	0.91
Cat	0.96	0.59	0.73	0.78	**Goodnight**	0.96	0.59	0.73	0.78	**Large**	0.96	0.63	0.76	0.79
Cop	0.91	0.91	0.91	0.95	**goout**	0.88	0.85	0.87	0.91	**Sorry**	0.85	1.00	0.92	0.95
Cost	0.85	0.79	0.81	0.87	**Hearing**	0.85	1.00	0.92	0.95	**Tiger**	0.58	0.64	0.61	0.76
Day	0.96	0.80	0.87	0.91	**Hello**	0.96	0.81	0.88	0.91	**Average**	0.86	0.82	0.83	0.88
Deaf	0.56	0.88	0.68	0.83	**Help**	0.88	1.00	0.93	0.96					

types of feedback and the possibility of peer-to-peer feedback for on-line learning has to be evaluated as suggested by works such as [10]. This work provides the foundations and feasibility for interactive and intelligent sign language learning to pave the path for such future work.

We collected usage and interaction data from 100 new learners as part of this work, which will be foundational to assist future researchers. Although, the focus of this work was on the manual portion of sign languages, the preprocessing includes location estimates for the eyes, ears and the nose. This can be utilized for including facial expression recognition and feedback in future works. We evaluated only 25 isolated words for ASL, but in the future, this work can be extended to more words and phrases and to include other sign languages since the general principles will remain the same. In this work, we used sign language as a test application, however, the insights from this work can be easily applied to other gesture domains such as combat sign training for military or industrial operator signs.

6 CONCLUSION

There is an increasing need and demand for learning sign language. Feedback is very important for language learning and intelligent language learning softwares must provide effective and meaningful feedback. There has also been significant advances in research for recognizing sign languages, however technological solutions that leverage them to provide intelligent learning environments do not exist. In this work, we identify different types of potential feedback we can provide to learners of sign language and address some challenges in doing so. We propose a pipeline of three non-parametric recognition modules and an incremental feedback mechanism to facilitate learning. We tested our system on real-world data from a variety of devices and settings to achieve a final recognition accuracy of 87.9 %. This demonstrates that using explainable machine learning for gesture learning is desirable and effective. We also provided different types of feedback mechanisms based on results of a user survey and best practices in implementing them. Finally, we collected data from 100 users of L2S with 3 repetitions for each of the 25 signs for a total of 7500 instances [13].

7 ACKNOWLEDGMENTS

We thank SigningSavvy[24] for letting us use their tutorial videos in the application.

REFERENCES

[1] Xavier Anguera, Robert Macrae, and Nuria Oliver. 2010. Partial sequence matching using an unbounded dynamic time warping algorithm. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 3582–3585.

[2] Modern Language Association. 2016. Language Enrollment Database. https://apps.mla.org/flsurvey_search. [Online; accessed 24-September-2018].

[3] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. 2016. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *European Conference on Computer Vision*. Springer, 561–578.

[4] Ching-Hang Chen and Deva Ramanan. 2017. 3d human pose estimation= 2d pose estimation+ matching. In *CVPR*, Vol. 2. 6.

[5] Xianjie Chen and Alan L Yuille. 2014. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *Advances in neural information processing systems*. 1736–1744.

[6] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural Sign Language Translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7784–7793.

[7] Konstantinos G Derpanis, Matthieu Lecce, Kostas Daniilidis, and Richard P Wildes. 2012. Dynamic scene understanding: The role of orientation features in space and time in scene classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 1306–1313.

[8] Farzad Ehsani and Eva Knodt. 1998. Speech technology in computer-aided language learning: Strengths and limitations of a new CALL paradigm. (1998).

[9] Karen Emmorey. 2001. *Language, cognition, and the brain: Insights from sign language research*. Psychology Press.

[10] Rebecca Fiebrink, Perry R Cook, and Dan Trueman. 2011. Human model evaluation in interactive supervised learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 147–156.

[11] Kirsti Grobel and Marcell Assan. 1997. Isolated sign language recognition using hidden Markov models. In *Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on*, Vol. 1. IEEE, 162–167.

[12] Pradeep Kumar, Himaanshu Gauba, Partha Pratim Roy, and Debi Prosad Dogra. 2017. Coupled HMM-based multi-sensor data fusion for sign language recognition. *Pattern Recognition Letters* 86 (2017), 1–8.

[13] Impact Lab. 2018. Learn2Sign Details Page. https://impact.asu.edu/projects/sign-language-recognition/learn2sign

[14] Kian Ming Lim, Alan WC Tan, and Shing Chiang Tan. 2016. A feature covariance matrix with serial particle filter for isolated sign language recognition. *Expert Systems with Applications* 54 (2016), 208–218.

[15] Malek Nadil, Feryel Souami, Abdenour Labed, and Hichem Sahbi. 2016. KCCA-based technique for profile face identification. *EURASIP Journal on Image and Video Processing* 2017, 1 (2016), 2.

[16] World Health Organization. 2018. Deafness and hearing loss. http://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss. [Online; accessed 24-September-2018].

[17] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. 2017. Towards accurate multi-person pose estimation in the wild. In *CVPR*, Vol. 3. 6.

[18] Prajwal Paudyal, Ayan Banerjee, and Sandeep KS Gupta. 2016. Sceptre: a pervasive, non-invasive, and programmable gesture recognition technology. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*. ACM, 282–293.

[19] Prajwal Paudyal, Junghyo Lee, Ayan Banerjee, and Sandeep KS Gupta. 2017. Dyfav: Dynamic feature selection and voting for real-time recognition of fingerspelled alphabet using wearables. In *Proceedings of the 22nd International*

Conference on Intelligent User Interfaces. ACM, 457–467.

[20] Fabrizio Pedersoli, Sergio Benini, Nicola Adami, and Riccardo Leonardi. 2014. XKin: an open source framework for hand pose and gesture recognition using kinect. *The Visual Computer* 30, 10 (2014), 1107–1122.

[21] Martha C Pennington and Pamela Rogerson-Revell. 2019. Using Technology for Pronunciation Teaching, Learning, and Assessment. In *English Pronunciation Teaching and Research*. Springer, 235–286.

[22] Sean Robertson, Cosmin Munteanu, and Gerald Penn. 2018. Designing Pronunciation Learning Tools: The Case for Interactivity against Over-Engineering. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 356.

[23] Russell S Rosen. 2010. American sign language curricula: A review. *Sign Language Studies* 10, 3 (2010), 348–381.

[24] Signing Saavy. 2018. Signing Saavy: Your Sign Language Resouce. https://www.signingsavvy.com/. [Online; accessed 28-September-2018].

[25] Nikolaos Sarafianos, Bogdan Boteanu, Bogdan Ionescu, and Ioannis A Kakadiaris. 2016. 3d human pose estimation: A review of the literature and analysis of covariates. *Computer Vision and Image Understanding* 152 (2016), 1–20.

[26] YoungHee Sheen. 2004. Corrective feedback and learner uptake in communicative classrooms across instructional settings. *Language teaching research* 8, 3 (2004), 263–300.

[27] Peter Skehan. 1998. *A cognitive approach to language learning*. Oxford University Press.

[28] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* (2012).

[29] Thad Starner, Joshua Weaver, and Alex Pentland. 1998. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on pattern analysis and machine intelligence* 20, 12 (1998), 1371–1375.

[30] William C Stokoe Jr. 2005. Sign language structure: An outline of the visual communication systems of the American deaf. *Journal of deaf studies and deaf education* 10, 1 (2005), 3–37.

[31] Rosetta Stone. 2016. Talking back required. https://www.rosettastone.com/speech-recognition. [Online; accessed 28-September-2018].

[32] Denis Tome, Christopher Russell, and Lourdes Agapito. 2017. Lifting from the deep: Convolutional 3d pose estimation from a single image. *CVPR 2017 Proceedings* (2017), 2500–2509.

[33] Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. 2014. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems*. 1799–1807.

[34] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*. 4489–4497.

[35] Fabienne M Van der Kleij, Remco CW Feskens, and Theo JHM Eggen. 2015. Effects of feedback in a computer-based learning environment on students' learning outcomes: A meta-analysis. *Review of educational research* 85, 4 (2015), 475–511.

[36] Ping Yu, Yingxin Pan, Chen Li, Zengxiu Zhang, Qin Shi, Wenpei Chu, Mingzhuo Liu, and Zhiting Zhu. 2016. User-centred design for Chinese-oriented spoken english learning system. *Computer Assisted Language Learning* 29, 5 (2016), 984–1000.