Sign Language Fingerspelling Recognition using Synthetic Data

Frank Fowley[1,2] and Anthony Ventresque[1]

[1] School of Computer Science, University College Dublin
[2] SFI Centre for Research Training in Digitally-Enhanced Reality (D-REAL)

Abstract. Sign Language Recognition (SLR) is a Computer Vision (CV) and Machine Learning (ML) task, with potential applications that would be beneficial to the Deaf community, which includes not only deaf persons but also hearing people who use Sign Languages. SLR is particularly challenging due to the lack of training datasets for CV and ML models, which impacts their overall accuracy and robustness. In this paper, we explore the use of synthetic images to augment a dataset of fingerspelling signs and we evaluate whether this could be used to reliably increase the performance of an SLR system. Our model is based on a pretrained convolutional network, fine-tuned using synthetic images, and tested using a corpus dataset of real recordings of native signers. An accuracy of 71% recognition was achieved using skeletal wireframe image training datasets and using the MediaPipe pose estimation model in the test pipeline. This compares favourably with state-of-the-art CV models which achieve up to 62% accuracy with "in-the-wild" fingerspelling test datasets.

Keywords: Sign Language Recognition · Synthetic Data · Data Augmentation · Convolutional Neural Network · Pose Estimation Model

1 Introduction

Deaf advocacy organizations maintain that the use of Sign Languages is a core right and can ensure that deaf people fully participate in society at large [1, 13]. However, in Ireland, the low number of Irish Sign Language (ISL) interpreters has led to their use being confined to important event contexts [4]. To address this issue, the development of a practical automated real-time ISL interpreter could have many applications in areas such as public service information, the Internet and social media, and in transport and medical contexts. Mobile and cloud-based interpreting applications and services could lead to increased freedom of expression, equal access to education and employment, as well as participation in cultural, sporting and entertainment activities [4].

Because of their greater degrees of articulatory freedom, Sign Languages have a richer and more complex phonology than spoken languages [11, 12]. In this paper, we will focus on fingerspelling, which is used not only for spelling out proper names, place names and abbreviations, but also as hand shapes for signs, such as the "m" hand shape used in the articulation of the "mother" sign.

There are 23 static letter signs and 3 moving letter signs ("J", "X" and "Z") in the ISL fingerspelling alphabet which is shown in figure 1.

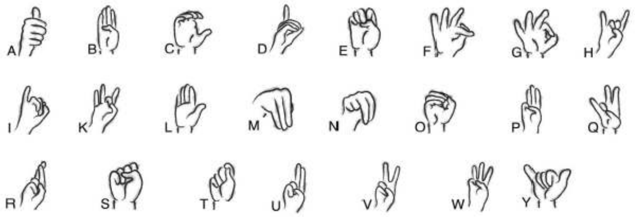


Fig. 1. The ISL Fingerspelling Alphabet - Static signs. (Source: Irish Deaf Society)

The Computer Vision (CV) task of recognising these fingerspelling signs is challenging due to the high degree of variation in signer fluency and linguistic effects such as co-articulation. SLR systems require a high degree of distortion invariance while maintaining high accuracy and low misclassification rates. Convolutional Neural Networks (CNNs) have proved useful in extracting features in images even with geometric transformations of the objects [10] but rely heavily on large datasets to avoid over-fitting. The effect of over-fitting is the degradation in model performance when applied to unseen test data. One of the fundamental challenges here when applying CV/ML to SLR is *a sparsity of datasets and corpora content, of sufficient scale and format to be useful as training input*[6]. This is partially due to the absence of written forms of Sign Languages and the costly nature of annotation [25].

Classical deep learning mitigation techniques can be used to address this issue: data augmentation - which improves the generalisation of neural networks and aims to expand the training dataset; and transfer learning [22, 24] - which refers to the use of a deep neural network, already pre-trained on a dataset, to be fine-tuned with a new dataset with a new set of classes [18].

In this paper, We describe a *novel method to overcome the lack of ISL training data by generating synthetic images at scale, applying transfer learning techniques to leverage the feature extraction capabilities of popular CNNs, and deploying current pose estimation models in the recognition process.*

The paper is structured as follows: Section 2 details some related work, before describing our research methods in Section 3. The experimental results are outlined in Section 4 and we conclude with a discussion of the results and future directions in Section 5.

2 Related Work

Many classical approaches have been used for sign and posture recognition, based on statistical pattern recognition and other shallow-learning techniques that require the initial definition of object features [13]. Farouk et al. [44] used synthetic images of ISL fingerspelling hand shapes to create a recognition model based on Principal Component Analysis (PCA). Experiments based on intrusive motion-capture equipment, such as gloves and wearable sensors, have been performed to create Sign Language interpreters [2], but they are of limited interest given the holistic nature of Sign Languages which includes features such as facial expressions, for instance. We have not included results from works based on plethysmography or electromyographic techniques as their use is considered by the Deaf community to be intrusive to the signer and therefore impractical [6, 41, 42].

Most published results for Sign Language fingerspelling recognition, including ISL, have been obtained in controlled environments where the training and test data are derived from the same subjects and in similar data capture conditions. The performance degradation of such models, when applied to unseen and "real-world" domains, known as "domain adaptation", is well documented [43]. To overcome this, our work is focussed on the more challenging scenario where there is separation between the training and test domains. The state-of-the-art figures cited in this paper were obtained from works that were conducted on non-controlled and "in-the-wild" settings.

Sign Language Technology In their systematic survey of Sign Language computational research, Zeledón et al.[15], highlight the low availability of commercial applications for Sign Language translation, confined in the main to synthesis rather than recognition, and restricted to particular domains. They report that the performance measures for state-of-the-art machine translation systems are between 70% and 80% (BLEU score)[3] and between 20% and 30% (WER score)[4] citing the limited availability of well-defined Sign Language grammars as a reason for the low performance in comparison to spoken language machine translation systems, as well as the challenge of properly annotating corpus recordings. Rastgoo et al. also state that the current state-of-the-art is focused on phonetic sign recognition rather than the lexicological and semantic areas which require more complex models [15].

Bragg et al., maintain that Deaf contributors should be involved at all facets of research and development in order to "accurately represent the community, address meaningful problems, and avoid cultural appropriation" [6]. They report that most signing datasets are only partially annotated. Continuous sign recognition is the most challenging part of the translation pipeline due to the complex phonology of signing, the variance in the fluency, dexterity, age and gender of the signer, the use of slang and dialect as well as the issues of occlusion and camera quality.

Deep Learning for SLR There has been a significant body of research published on the application of deep learning techniques to SLR [13, 20]. Shi et

[3] BLEU (Bilingual Evaluation Understudy) is a standard machine translation evaluation calculation method.

[4] WER (Word Error Rate) is a measure of the changes needed in the words of a phrase to transform it into another phrase.

al. [23] achieved state-of-the-art accuracy of 62.3% recognition on a dataset of "in-the-wild" videos of American Sign Language (ASL) fingerspelling, using an attention-based recurrent neural network. Halvardsson et al. [8] apply transfer learning techniques to three CNNs (InceptionRes-NetV2 [33], Xception [32] and InceptionV3 [31]) to recognise static manual signs of the Swedish Sign Language finger-spelling alphabet. They obtain 85% accuracy using the InceptionV3 network with 5 fine-tuned layers, on a test dataset derived from 8 recordings of 6 subjects. Though signer-independent, their training and test datasets were created in the same controlled environment. They demonstrate that the accuracy is dependent on the number of pre-trained layers.

Synthetic Data and Transfer Learning Transfer Learning techniques have been applied to SLR with encouraging results using several popular pre-trained deep learning networks and configurations [8, 14]. Synthetic data, produced from artificial means rather than by human photography, has been used to train ML models for CV applications, to train generative models, to augment real data datasets and to anonymise real data in privacy sensitive scenarios [5]. Techniques using synthetic data have been applied to problems in object detection and segmentation, face and text recognition, image classification and pose estimation [16].

Nikolenko's review [16] suggests that best results are obtained when combining synthetic datasets from different domains. Bayraktar et al. [3] use synthetic data to fine-tune the VGGNet [37], Inception, ResNet [36] and Xception neural networks in object detection experiments and report that a mix of real and synthetic data yields best results. Peng et al. [19] show that texture and colour variations in training datasets are more important than pose variations. Hinterstoisser et al. [9] present a similar experimental technique to the one presented in this paper. They propose to retain the feature extraction of lower layers in the networks deployed, only fine-tuning the higher order blocks with synthetic data. Rajpura et al. [21] use synthetic images, generated by Blender, and transfer learning to fine-tune three CNNs (DetectNet, Faster R-CNN and SSD) to create a network to recognise a set of household objects and more than doubled the model's precision score. They report that class set size and fine-tuning depth have a significant effect on performance and that the optimal accuracy is obtained by fine-tuning all of the underlying DetectNet Inception layers. Goyal et al. [7] use augmented synthetic data for segmentation models, fine-tuning the top-most five layers of a CNN with Blender-rendered synthetic images. They show significant precision score improvements after retraining the FNC-8s network with a subset of the PASCAL dataset [39], fine-tuning the result with a small synthetic dataset. They report that the model's success depends on class sample size and object type. They limit the fine-tuning depth of their experiments to account for the non photo-realism of the synthetic images. There have been some studies which have implemented pose estimation models for Sign Language recognition, mainly using the joint coordinates from the OpenPose [35] model as input data [26–29].

Contributions Our approach differs from the previous research by using larger synthetic datasets than those available in Sign Language corpora. Through the use of an automated framework, we can control the variations within the training dataset and can generate ground-truth frame-level annotation automatically. Furthermore, we adopt a pose estimation model in the recognition pipeline to reduce the domain shift between training and test datasets. We use customised wireframe skeletal images to exploit the performance of current CNN models through transfer learning techniques.

3 Methodology

3.1 Programmable Pose Framework

Our approach includes the use of data augmentation to include translational, viewpoint, size and illumination invariance into the training datasets to enable the model to overcome illumination, camera perspective, background, anatomical and pose variations found in real-world scenarios. We developed a framework to automate the generation of hand poses. It is based on a skeletal-rigged 3D hand avatar mesh loaded into the Blender graphics engine and can be programmed to produce synthetic data variations at scale. The skeletal armature of the hand model can be rotated and positioned by setting its constituent bones from a pre-set list of parameterised features such as “bent”, “hooked”, “curled”, etc. An individual ISL manual shape is composed of a set of these features which thus determines its pose. There is no hand-crafting or manual setting of the hand mesh required for the animation.

Figure 2 shows an example of pose variations where the finger rotations are varied to correspond to the differences in fluency of signers and phonetic variants found in ISL. The graphics engine allows for variations in scene illumination and camera perspective by setting the positions of cameras and lights. The extent of the image alterations and class balance can be controlled programmatically in the framework.

The system outputs colour, greyscale, depth and skeletal wireframe images and video, as well as skeletal joint key-point coordinates corresponding to the animated hand shapes. The wireframe images can be generated in the format of pose estimation models, OpenPose, MediaPipe [34] and Kinect4Azure [30]. Examples of colour and wireframe images generated by the framework are shown below.

3.2 Training Datasets

The experiments were divided into two phases. The first phase used synthetic RGB images of hands. The second phase introduced a pose estimation model into the pipeline and used images based on the output of this model as training data.

Phase 1: Hand RGB Images The Phase 1 experiments used training datasets of approximately 520,000 synthetic RGB images rendered in the synthetic framework, a sample of which is shown in figure 2.

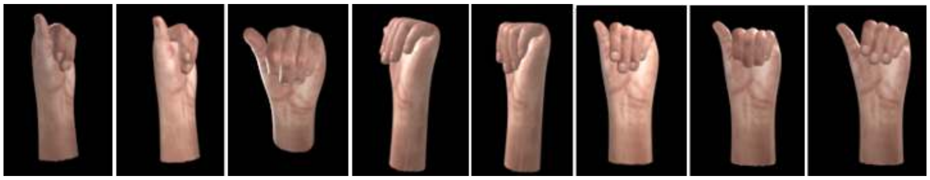


Fig. 2. Blender Synthetic Framework Examples. 10 synthetic images of ‘A’ from different camera perspectives and illumination conditions. Poses show thumb angle variances, finger curling, finger on palm, non-visible nails and random finger bone rotations.

Phase 2: Hand Skeletal Images The training dataset for the Phase 2 models was based on approximately 1.6 million pose wireframe images in the format produced by pose estimation models. We experimented with several modifications to the wireframe output formats to compare the effectiveness of different artificial features in the images. The logic is to add visual, geometric or pixel-based, potentially discriminable features to the training dataset images. Figure 3 shows two such “Feature Injection” formats generated by the Blender engine. The best performing training dataset was based on colouring the individual fingers with evenly spaced hues keeping all bones in the same finger the same colour without any pixel thickness change between fingers or bones. The MediaPipe API allows its wireframe images to be outputted with these above modifications at inference time.

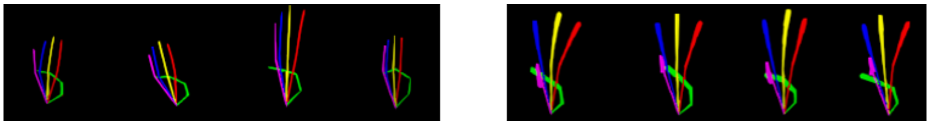


Fig. 3. Phase 2 Dataset Samples. MediaPipe WireFrames with Feature Injection. Left: 4 samples of “B” with different hue per finger. Right: 4 samples of “W” with different hue per finger and different thickness per bone.

3.3 Test Dataset

The models were tested using the ISL-HS corpus of fingerspelling signs [17]. This is a dataset of ISL fingerspell signs captured from 6 native ISL signers. They are composed of 3 recordings of each person, resulting in 468 videos of static ISL alphabet letters.

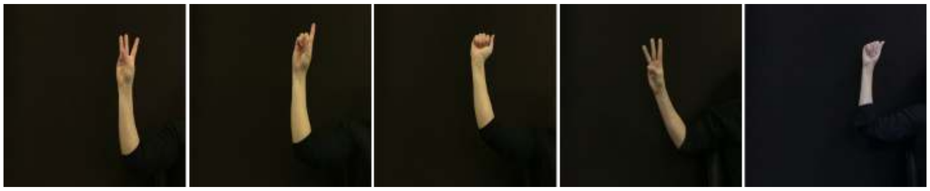


Fig. 4. Test Dataset - ISL-HS Corpus. Five samples showing letters 'W', 'D', 'A', 'W' and 'A', with different subjects using different finger poses for the same letter such as curled fingers and straight fingers on palm for two 'A' samples.

The corpus is available as a dataset of 52,688 grayscale images, with an average of approximately 2,290 samples per alphabet letter, for the 23 static signs. For the test datasets used in this work, we abstracted the RGB colour images from the video frames. Figure 4 shows a sample of the corpus images. For Phase 2 experiments, which use a pose model wireframe output as data, the pose estimation model was applied to the RGB ISL-HS images to create the wireframe test datasets.

3.4 Model Pipeline

The model training and evaluation pipelines are outlined in figure 5. Different open-source networks were used during the experiments including VGGNet16, InceptionV3, Xception, ResNet152V2 and MobileNetV2. The models were trained with a learning rate of 0.0001 and used the Adam optimiser. Batch-sizes of 16, 32 and 64 were compared in the hyper-parameter adjustments with 16 proving to be optimal. A training/validation set split of 90:10 was used throughout the experiments.

3.5 Frameworks and Equipment

The Blender framework and API were used to develop the synthetic data framework. The Keras TensorFlow framework was deployed for all the deep learning pipeline tasks, along with the OpenCV2 and MediaPipe APIs for data preprocessing. Python Jupyter Notebooks were used for all experimental coding. No Keras data augmentation functions were used since it is the role of the synthetic image framework to control the data augmentation. The experiments were carried out on a Dell Precision 5820 Tower WorkStation equipped with an Intel Xeon W-2235 Processor and 32GB CPU RAM and fitted with an NVIDIA Quadro RTX5000 GPU with 16GB GPU RAM.

4 Experimental Results

The accuracy of the best performing model in Phase 1, trained on RGB images, is 33% overall, using a VGGNet16 network with its top convolutional block

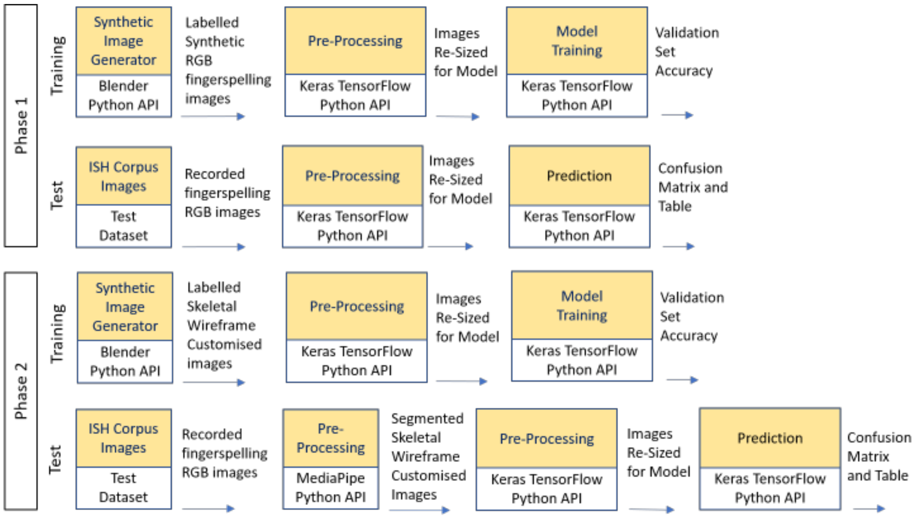
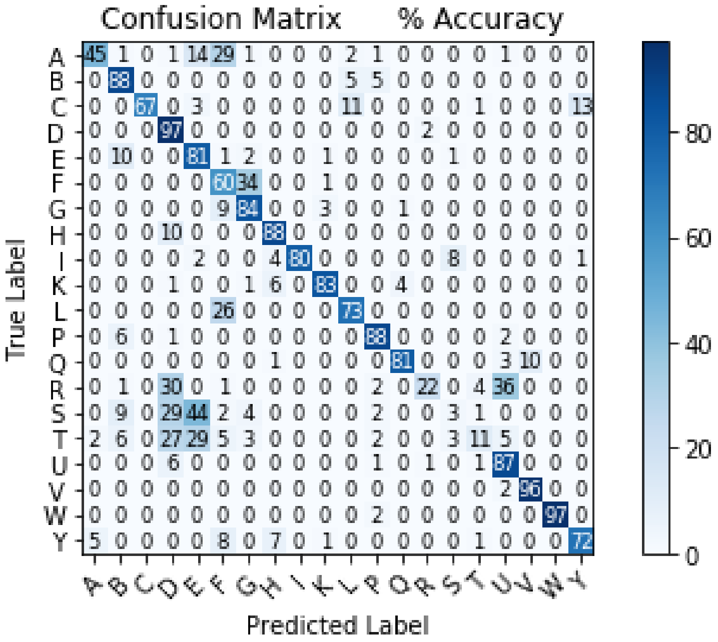

Phase 1
Training
Synthetic Image Generator
Blender Python API
Labelled Synthetic RGB fingerspelling images
Pre-Processing
Keras TensorFlow Python API
Images Re-Sized for Model
Model Training
Keras TensorFlow Python API
Validation Set Accuracy
Test
ISH Corpus Images
Test Dataset
Recorded fingerspelling RGB images
Pre-Processing
Keras TensorFlow Python API
Images Re-Sized for Model
Prediction
Keras TensorFlow Python API
Confusion Matrix and Table
Phase 2
Training
Synthetic Image Generator
Blender Python API
Labelled Skeletal Wireframe Customised images
Pre-Processing
Keras TensorFlow Python API
Images Re-Sized for Model
Model Training
Keras TensorFlow Python API
Validation Set Accuracy
Test
ISH Corpus Images
Test Dataset
Recorded fingerspelling RGB images
Pre-Processing
MediaPipe Python API
Segmented Skeletal Wireframe Customised Images
Pre-Processing
Keras TensorFlow Python API
Images Re-Sized for Model
Prediction
Keras TensorFlow Python API
Confusion Matrix and Table

Fig. 5. Training, Evaluation and Inference Pipelines.

re-trained. While the model's confusion matrix showed some encouraging class recognition, the results also revealed significant confusion between some classes. These subsets of non-discriminatory classes appeared in experiments with varying hyper-parameter settings. This suggests that some letters are inherently more difficult to discriminate, which reflects the actual physical similarity between some fingerspelling shapes.

To enable the models to discriminate between these, we used wireframe images in the format of pose estimation model output as our training datasets in Phase 2 (This would subsequently require the use of a pose estimation model in the test pipeline). The synthetic pose framework was extended to generate training datasets of skeletal wireframe images in the requisite format. We then manipulated the visual aspects of the images such as the colours and widths of bones and fingers, and trained our model based on these synthetic training datasets. The best performing model in Phase 2 used a VGGNet16 base network. For our tests, we used the MediaPipe pose estimation model in the test pipeline. The Phase 2 results yielded an overall recognition accuracy of 71.4% when applied to a corpus test dataset of recorded ISL fingerspelling alphabets (These images having been pre-processed into wireframe images by applying the pose model.)

The optimal fine-tuning depth in Phase 2 is greater than that of Phase 1, with a re-training of three convolutional blocks yielding best results. There was a rise in accuracy of 4.7 percentage points resulting from a three-fold increase in the size of the training dataset. The results from the best-performing VGGNet16 configuration are shown in figure 6. However, there is still a marked confusion between 'E' and 'S', 'G' and 'F', and 'R' and 'U', as seen in table 1. This confusion table shows the accuracy of the model as the percentage of correctly



	A	B	C	D	E	F	G	H	I	K	L	P	Q	R	S	T	U	V	W	Y
A	45	1	0	1	14	29	1	0	0	0	2	1	0	0	0	0	1	0	0	0
B	0	88	0	0	0	0	0	0	0	0	5	5	0	0	0	0	0	0	0	0
C	0	0	67	0	3	0	0	0	0	0	11	0	0	0	0	1	0	0	0	13
D	0	0	0	97	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
E	0	10	0	0	81	1	2	0	0	1	0	0	0	0	1	0	0	0	0	0
F	0	0	0	0	0	60	34	0	0	1	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	9	84	0	0	3	0	0	1	0	0	0	0	0	0	0
H	0	0	0	10	0	0	0	88	0	0	0	0	0	0	0	0	0	0	0	0
I	0	0	0	0	2	0	0	4	80	0	0	0	0	0	8	0	0	0	0	1
K	0	0	0	1	0	0	1	6	0	83	0	0	4	0	0	0	0	0	0	0
L	0	0	0	0	0	26	0	0	0	0	73	0	0	0	0	0	0	0	0	0
P	0	6	0	1	0	0	0	0	0	0	0	88	0	0	0	0	2	0	0	0
Q	0	0	0	0	0	0	0	1	0	0	0	0	81	0	0	0	3	10	0	0
R	0	1	0	30	0	1	0	0	0	0	0	2	0	22	0	4	36	0	0	0
S	0	9	0	29	44	2	4	0	0	0	0	2	0	0	3	1	0	0	0	0
T	2	6	0	27	29	5	3	0	0	0	0	2	0	0	3	11	5	0	0	0
U	0	0	0	6	0	0	0	0	0	0	0	1	0	1	0	1	87	0	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	96	0	0
W	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	97	0
Y	5	0	0	0	0	8	0	7	0	1	0	0	0	0	0	1	0	0	0	72

Fig. 6. Phase 2 Results. Confusion Matrix. Overall Accuracy 71.4%.

Table 1. Accuracy for each letter with 3 most confused letters.

Letter	Accuracy	Top 3 Most Closely Confused Letters		
A	0.46	L (0.02)	E (0.15)	F (0.29)
B	0.89	E (0.01)	L (0.05)	P (0.05)
C	0.68	E (0.04)	L (0.12)	Y (0.13)
D	0.97	E (0.00)	T (0.00)	R (0.02)
E	0.81	F (0.01)	G (0.02)	B (0.10)
F	0.61	Q (0.01)	G (0.35)	K (0.02)
G	0.85	Q (0.01)	F (0.09)	K (0.04)
H	0.89	B (0.01)	T (0.01)	D (0.10)
I	0.81	E (0.03)	S (0.09)	H (0.04)
K	0.83	G (0.02)	Q (0.04)	H (0.07)
L	0.74	I (0.00)	H (0.00)	F (0.26)
P	0.88	D (0.01)	U (0.03)	B (0.07)
Q	0.82	H (0.01)	U (0.03)	V (0.10)
R	0.23	T (0.04)	U (0.36)	D (0.31)
S	0.04	E (0.45)	D (0.29)	B (0.09)
T	0.12	B (0.06)	D (0.27)	E (0.29)
U	0.88	P (0.01)	R (0.01)	D (0.06)
V	0.97	D (0.00)	E (0.01)	U (0.02)
W	0.97	Q (0.00)	B (0.00)	P (0.02)
Y	0.72	A (0.05)	H (0.08)	F (0.09)
Overall Accuracy	0.71			

recognised test samples for any letter. It also shows the three letters that have been incorrectly classified by the model for any class, in terms of the highest percentage of test samples. For example, although the model correctly recognised 'F' in 61% of test samples, the model also incorrectly classified 35% of 'F' test samples as 'G', 1% as 'Q' and 2% as 'K', thus showing scope for further potential improvements. They are, in effect, the three letters most "confused" by the model when recognising an 'F'. Details of the above experimental results as well as all corresponding code have been made available on GitHub. [5]

5 Conclusion and Future Work

The above results demonstrate that a CNN, trained solely on synthetic images, can effectively recognise isolated ISL fingerspelling signs. There is a need to resolve the recognition confusion evident in a small subset of classes with techniques such as ensemble learning and composite models. We plan to extend the synthetic image generator and the recognition models to cater for the full set of ISL handshapes as well as dynamic signs and to eventually recognise continuous sign sequences. The latter will require the models to be extended with a temporal architecture such as a recurrent neural network (RNN) or LSTM [40] structure. We hope to create a corpus of native ISL signer recordings, in formats suitable for input to the deep learning models, as well as a database of annotated ISL online videos as a comprehensive "in-the-wild" test dataset. While occlusion was not a problem for one-handed fingerspelling recognition, paired synchronised depth sensors will be deployed in future pipelines with appropriate models to cater for this effect.

Acknowledgement

This work was conducted with the financial support of the Science Foundation Ireland Centre for Research Training in Digitally-Enhanced Reality (D-REAL) under Grant No. 18/CRT/6224. This work has also been conducted within the SignON project. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017255.

References

1. EUD Homepage, https://www.eud.eu/about-us/eud-position-paper/accessibility-information-and-communication/. Last checked 05.08.2021.
2. Mohamed Aktham Ahmed, Bilal Bahaa Zaidan, Aws Alaa Zaidan, Mahmood Maher Salih, and Muhammad Modi Bin Lakulu.: A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017. In: Sensors, 18(7):2208, 2018.

[5] https://github.com/ucd-csl/ISL-SLR

3. Ertugrul Bayraktar, Cihat Bora Yigit, and Pinar Boyraz.: A hybrid image dataset toward bridging the gap between real and simulation environments for robotics. In: MVA 2019.
4. Citizens Information Board. Information provision and access to public and social services for the Deaf Community. Government of Ireland, December 2017. https://www.citizensinformationboard.ie/downloads/social_policy/. Last checked 05.08.2021.
5. Erik Bochinski, Volker Eiselein, and Tomas Sikora.: Training a convolutional neural network for multi-class object detection using solely virtual world data. In: AVSS 2016.
6. Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa Verhoef, Christian Vogler, and Meredith Ringel Morris: Sign language recognition, generation, and translation: An interdisciplinary perspective. In: ASSETS 2019.
7. Manik Goyal, Param Rajpura, Hristo Bojinov, and Ravi Hegde.: Dataset augmentation with synthetic images improves semantic segmentation. Communications. In: NCVPRIPG 2018.
8. Gustaf Halvardsson, Johanna Peterson, C. Soto-Valero, and Benoit Baudry.: Interpretation of swedish sign language using convolutional neural networks and transfer learning. In: SN 2021.
9. Stefan Hinterstoisser, Vincent Lepetit, Paul Wohlhart, and Kurt Konolige.: On pre-trained image features and synthetic images for deep learning. In: ECCV 2018 Workshops, page 682–697, 2019.
10. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner.: Gradient-based learning applied to document recognition. In: Proceedings of the IEEE, 1998.
11. L. Leeson and J.I. Saeed.: Irish Sign Language: A Cognitive Linguistic Account. Edinburgh University Press, 2012.
12. Patrick A. Matthews.: Extending the Lexicon of Irish Sign Language (ISL) [microform] / Patrick A. Matthews. Distributed by ERIC Clearinghouse [S.l.], 1996.
13. Z. Omar Ming Jin Cheok and M. Jaward.: A review of hand gesture and sign language recognition techniques. In: International Journal of Machine Learning and Cybernetics, 10:131–153, 2019.
14. Boris Mocialov, Graham Turner, and Helen Hastie.: Transfer learning for british sign language modelling, 2020.
15. Luis Naranjo-Zeled´on, Jes´us Peral, Antonio Ferr´andez, and Mario Chac´on-Rivas.: A systematic mapping of translation-enabling technologies for sign languages. In: Electronics, 8(9), 2019.
16. Sergey I. Nikolenko.: Synthetic data for deep learning, 2019.
17. Marlon Oliveira, Houssem Chatbri, Suzanne Little, Ylva Ferstl, Noel E. Oconnor, and Alistair Sutherland.: Irish sign language recognition using principal component analysis and convolutional neural networks. In: DICTA 2017.
18. Sinno Jialin Pan and Qiang Yang.: A survey on transfer learning. In: TKDE, 22(10):1345–1359, 2010.
19. Xingchao Peng, Baochen Sun, Karim Ali, and Kate Saenko.: Exploring invariances in deep convolutional neural networks using synthetic images. 2014.
20. R. Rastgoo, K. Kiani and S. Escalera.: Sign language recognition: A deep survey. In: ESA, 164:113794, 2021.
21. Param Rajpura, Alakh Aggarwal, Manik Goyal, Sanchit Gupta, Jonti Talukdar, Hristo Bojinov, and Ravi Hegde.: Transfer learning by finetuning pretrained cnns entirely with synthetic images. In: NCVPRIPG, page 517–528, 2018.

22. Ling Shao, Fan Zhu, and Xuelong Li.: Transfer learning for visual categorization: a survey. In: TNNLS 2015.
23. Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich, and Karen Livescu.: Fingerspelling recognition in the wild with iterative visual attention. In: CoRR, abs/1908.10546, 2019.
24. Karl Weiss, Taghi Khoshgoftaar, and DingDing Wang.: A survey of transfer learning. In: Journal of Big Data, 3, 05 2016.
25. L. Leeson, J. Saeed, and D. Byrne-Dunne. : Moving heads and moving hands Developing a digital corpus of irish sign language. 2006.
26. Bowen Shi, Diane Brentari, Greg Shakhnarovich and Karen Livescu : Fingerspelling Detection in American Sign Language. In: CVPR 2021.
27. Dongxu Li and Cristian Rodriguez-Opazo and Xin Yu and Hongdong Li : Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison. In: WACV 2020.
28. Hamid Reza Vaezi Joze and Oscar Koller : MS-ASL: A Large-Scale Data Set and Benchmark for Understanding American Sign Language. In: WACV 2020.
29. Sang-Ki Ko, Jae Gi Son, and Hyedong Jung. : Sign Language Recognition with Recurrent Neural Network using Human Keypoint Detection. In: RACS 2018.
30. kinect, https://azure.microsoft.com/en-us/services/kinect-dk/.
31. Christian Szegedy and Wei Liu and Yangqing Jia and Pierre Sermanet and Scott Reed and Dragomir Anguelov and Dumitru Erhan and Vincent Vanhoucke and Andrew Rabinovich : Going Deeper with Convolutions. 2014.
32. François Chollet : Xception: Deep Learning with Depthwise Separable Convolutions. 2017.
33. Christian Szegedy and Sergey Ioffe and Vincent Vanhoucke and Alex Alemi : Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. 2016.
34. mediapipe, https://google.github.io/mediapipe/. Last checked
35. Zhe Cao and Gines Hidalgo and Tomas Simon and Shih-En Wei and Yaser Sheikh : OpenPose. Realtime Fields. In: CoRR, 2018.
36. Kaiming He and Xiangyu Zhang and Shaoqing Ren and Jian Sun: Deep Residual Learning for Image Recognition. 2015.
37. Karen Simonyan and Andrew Zisserman : Very Deep Convolutional Networks for Large-Scale Image Recognition. 2015.
38. Liu, Wei and Anguelov, Dragomir and Erhan, Dumitru and Szegedy, Christian and Reed, Scott and Fu, Cheng-Yang and Berg, Alexander C. : SSD Single Shot MultiBox Detector. 2016. In: LNCS 2016.
39. Mark Everingham and Luc Van Gool and Christopher K. I. Williams and John M. Winn and Andrew Zisserman. : The Pascal Visual Object Classes (VOC)
40. Hochreiter, Sepp and Schmidhuber, Jürgen. : Long Short-term Memory. In: Neural computation. 1997.
41. Why Sign-Language Gloves Don't Help Deaf People. The Atlantic 9 (2017), https://www.theatlantic.com/technology/archive/2017/. Last checked 28.11.2021
42. Those Signing Gloves Are Not That Great. Language First 2019, https://language1st.org/essays/2019/6/15/those-signing-gloves-are-not-that-great. Last checked 28.11.2021
43. Charles, J, T Pfister, D Magee, D Hogg, and A Zisserman. 2013 : Domain Adaptation for Upper Body Pose Tracking in Signed TV Broadcasts. In: BMVA 2013.
44. Farouk, Mohamed, Sutherland, Alistair and Shoukry, Amin A. (2013) : Nonlinearity reduction of manifolds using Gaussian blur for handshape recognition based on multi-dimensional grids. In: ICPRAM 2013.