Sign-to-Speech Model for Sign Language Understanding: A Case Study of Nigerian Sign Language

Steven Kolawole[1*], **Opeyemi Osakuade**[2], **Nayan Saxena**[1] and **Babatunde Kazeem Olorisade**[3]
[1]ML Collective
[2]Data Science Nigeria
[3]Cardiff Metropolitan University
steven@mlcollective.org, osakuade@datasciencenigeria.ai, nayan@mlcollective.org,
kolorisade@cardiffmet.ac.uk

Abstract

Through this paper, we seek to reduce the communication barrier between the hearing-impaired community and the larger society who are usually not familiar with sign language in the sub-Saharan region of Africa with the largest occurrences of hearing disability cases, while using Nigeria as a case study. The dataset is a pioneer dataset for the Nigerian Sign Language and was created in collaboration with relevant stakeholders. We preprocessed the data in readiness for two different object detection models and a classification model and employed diverse evaluation metrics to gauge model performance on sign-language to text conversion tasks. Finally, we convert the predicted sign texts to speech and deploy the best performing model in a lightweight application that works in real-time and achieves impressive results converting sign words/phrases to text and subsequently, into speech.

1 Introduction

Communication has always been the mainstay of functional human interaction in any society. The hearing-impaired community uses sign language for effective interpersonal communication. Whilst the use of sign language works well as a means of communication between the hearing-impaired community, it is not the same while attempting to communicate with people outside of their community. In 2012, WHO estimated that over 5.3% of the world's population (about a 430 million) have hearing disabilities and it is most prevalent in sub-Saharan Africa [Organization and others, 2018]. Despite this prevalence, facilities to support their communication with the larger society are still lacking, most especially in developing countries. Children with hearing loss and deafness often do not receive formal schooling; adults with hearing loss suffer a higher unemployment rate; a higher percentage of those employed are in the lower grades of employment compared with the general workforce. There are also other impacts including social isolation, loneliness, and stigmatization. Although there has been an evolution of computer vision with artificial intelligence in creating innovations to solve hearing disabilities problems, we have very few of these solutions targeted towards developing countries. This is mostly due to two factors: (i) The sign language data in the region is low resourced (ii) There are increasing complexities and advanced tools required to deploy these solutions in real environments.

In this research project, we facilitate the creation of low-resource sign language datasets for countries where hearing impairments are most prevalent using the Nigeria Sign Language as a case study. A good dataset should sufficiently represent a challenging problem to make it both useful and to ensure its longevity – to the authors knowledge this dataset is the first of its kind for the low-resource sign languages in sub-Saharan Africa. The dataset images were annotated for object detection using LabelImg , in both YOLOv and PASCAL VOC formats. Furthermore, two object detection models and a classification model were trained and compared across multiple evaluation metrics. The results of this work clearly demonstrate that if provided the availability of otherwise low-resource data, the communication barrier between the hearing impaired community and the larger society can be bridged using the sign-to-speech machine learning techniques that can further be deployed to work in real-time. Link to demo is here.

1.1 Related Work

There is an abundance of American Sign Language datasets available publicly as a high-resource sign language, with the most recent one being a large-scale Word-Level American Sign Language dataset [Li *et al.*, 2020]. Sign languages just like spoken languages vary considerably from one location to another. Asides a masters thesis done on the South African Sign Language and a very small corpus of Ghanaian Sign Language which serves as a proof of concept [Odartey *et al.*, 2019], no further work has been done in creating standard datasets for sub-Saharan African sign languages.

2 Nigerian Sign Language Dataset

The dataset comprises of around 5000 images with 137 sign words, including the 27 alphabet letters. It should be noted that sign words and phrases are not just dependent on the signs being performed, but on the facial expression, the body posture, the relative position of the signs to the body, and motion. In fact, some signs are disambiguated based only on

[*] Contact Author

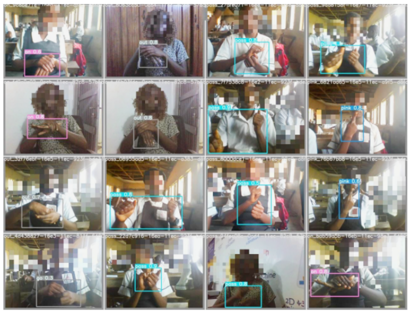

Figure 1: A mosaic of a fraction of the dataset in diverse backgrounds and lighting conditions.

motion making it challenging to create a dataset that is non-complex and easy to reuse. Hence, for signs that are heavily dependent on motion, we tried to capture the point in the motion where that moment is peculiar to the specific sign only. Supplementary materials, including code and model configs, are made available on Github here.

2.1 Dataset Creation

The dataset focuses on the Nigerian Sign Language and the initial data was created by a TV sign language broadcaster from the Ogun State Broadcasting Corporation. Further works to expand the dataset, with the aim of creating a wider dispersion in the dataset's distribution, were carried out in conjunction with teachers and students totalling 20 individuals from two special education schools in Abeokuta and Lagos– both cities in Nigeria. Figure 1 reflects the diverse backgrounds and lighting conditions in which the images were captured.The Nigerian Sign Language, like most of the other sign languages in the world, is quite influenced by the American Sign language, with a chunk of the language adopted from the British Sign Language and a smattering of it being vernacular signs. As shown in Figure 1, an initial 8000 static images of 137 signs words/phrases, including the 27 alphabet letters, were created with 20+ individuals in diverse environments, under different environments, to account for the different environments.

2.2 Preprocessing & Annotation

Data cleaning was performed to weed out excessively blurry images and images where the signs were not totally contained in the image frames. This reduced our images from 8000 to 5000 images. All images were resized into 640 x 640, therefore having a shape of (640, 640, 3). Images of the same class names were stored in the same image self-named folders formatted as "`classname_id.jpg`".A core part of computer vision tasks is to label the data, making it essential for the dataset to be annotated before it can be used with object detection models. This was done using LabelImg[2], a graphical image labeling tool, which was used to draw bounding boxes around the signs performed in each image. These rectangular bounding boxes define the location of the target object (sign in this case) with each bounding box consisting of the x and y coordinates (xmin-top left, ymin-top left, xmax-bottom right, ymax-bottom right) [Everingham *et al.*, 2010]. The annotations were created in both PASCAL VOC format and the YOLO format.

3 Modelling Experiments and Evaluation

We designed three models using two object detection architectures, YOLO and Single-Shot-Detector, and a fine-tuned pre-trained model for classification and compared results across the three models.

Evaluation Metrics The Precision is calculated as the ratio between the number of positive samples correctly classified to the total number of samples classified as positive (either correctly or incorrectly) while the recall is calculated as the ratio between the number of positive samples correctly classified as positive to the total number of positive samples. The recall measures the model's ability to detect positive samples and the precision measures the model's accuracy in classifying a sample as positive. A precision-recall (PR) curve shows the trade-off between the precision and recall values for different thresholds and the mAP is a way to summarize the PR-curve into a single value representing the average of all precisions. Training an object detection model usually requires two inputs which are the image and the ground-truth bounding boxes for the object in each image. When the model predicts the bounding box, it is expected that the predicted box will not match exactly the ground-truth box. IOU (Intersection over Union) is calculated by dividing the area of intersection between the two boxes by the area of their union which means that higher IOU scores generally translate into better predictions. Specifically, in this paper, we measure where there is a 50% overlap (@0.5) and where there is a 95% overlap (@0.95).

3.1 Object Detection Using YOLO

YOLO's unified architecture is an extremly fast architecture that reframes object detection as a single regression problem, straight from image pixels to bounding box coordinates and class probabilities. Using this architecture, you only look once (YOLO) at an image to predict what objects are present and where they are [Redmon *et al.*, 2016]. We made use of YOLOv5m implementation with the annotations in YOLO format and trained across 150 epochs. Data augmentation techniques including scaling, left-right flipping, HSV (Hue, Saturation, and Value) manipulation among others, were performed on the data. After the training process, we evaluated the performance of the model using several metrics including Precision, Recall, and mAP (mean Average Precision) when IOU are at 0.5 (50%) and 0.95 (95%). After evaluation, the YOLO model had a validation precision score of 0.8057, recall score of 0.95, as well as mAP scores of 0.95 and 0.64 for @0.5IOU and @0.95IOU respectively. This result confirms the effectiveness of our approach in predicting signs performed in diverse environments correctly.

3.2 Object Detection using Single-Shot Detector

The Single Shot Detector (SSD) architecture uses a single deep neural network in predicting category scores and box offsets for a fixed set and achieves high detection accuracy by producing predictions of different scales from feature maps of

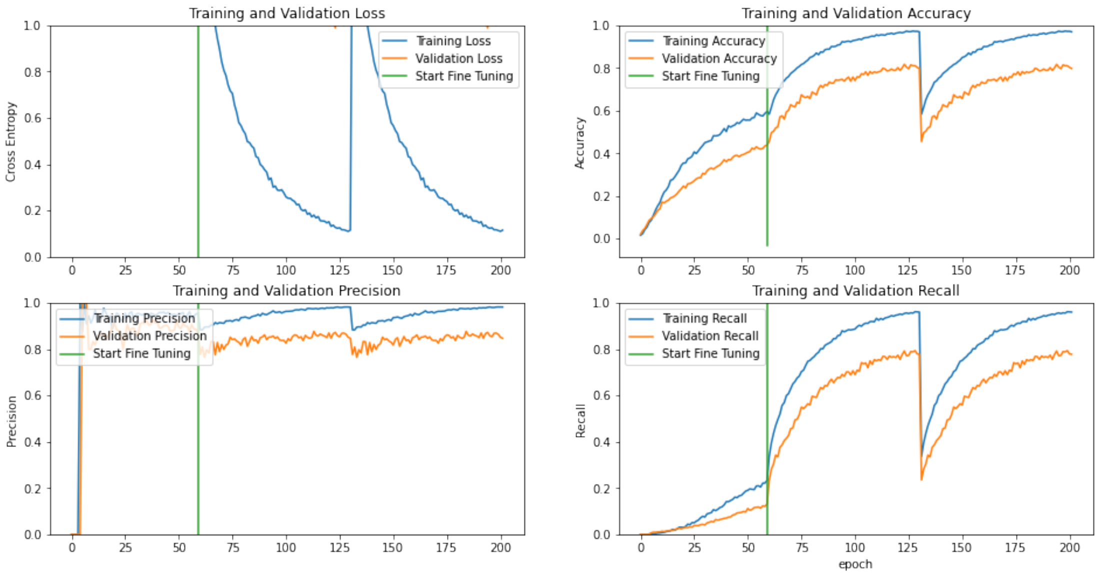

Figure 2: Graphs comparing the feature extraction performance with the fine-tuned performance.

different scales and explicitly separate predictions by aspect ratio [Liu *et al.*, 2016]. We used TensorFlow Object Detection API to access and finetune the SSD ResNet50 V1 FPN model, which was pre-trained on the COCO 2017 dataset [Lin *et al.*, 2014], and we fed the model our dataset which was converted into TensorFlow Record format. We made use of only horizontal flip and image cropping for Data augmentation, and we train across 40,000 train steps. Then we evaluate the model on the test set using Precision, Recall, and mAP metrics. After evaluation, the SSD model achieved a Precision score of 0.6414, a Recall score of 0.7075, mAP scores of 0.9535 and 0.6412 for 50% and 95% overlap respectively. This result is not as good as the YOLO model's result. Further investigations revealed that the SSD model is a little bit "fond" of imagining signs from random shapes. We deduced that the SSD model might require a much larger dataset to produce results on par with the YOLO model's result.

3.3 Classification using Transfer Learning

We decided to model the problem as a classification problem and train on the dataset while discarding the annotations. The images were resized to 224x224px. and applied to a pre-trained image classification MobileNet V2 model [Sandler *et al.*, 2018] which was then customized in 2 ways.

Feature Extraction on a Pretrained Model We used the representation learned by MobileNetV2 when pre-trained on the ImageNet dataset [Sandler *et al.*, 2018] to extract features from our data. We built a small model atop the pre-trained model without training any layers from the pre-trained model. Upon evaluation after 60 epochs, our model achieved a test accuracy score of 0.4397, precision score of 0.8640, and 0.1401 recall score.

Finetuning on the Pretrained Model The poor performance indicated that the model required fine-tuning. MobileNetV2 has 154 layers, hence we left only the first 100 layers frozen and train on both the newly-added classifier layers and the last layers of the pre-trained model. This helped fine-tune the higher-order representations in the base model thereby making it more relevant for our specific task. As demonstrated in Figure 2, we evaluated the model after training for an additional 140 epochs and the results were: 0.9115 accuracy score, 0.9355 precision score, and 0.9063 recall score.By finetuning the pretrained model, we reduced the loss sporadically and both the accuracy score and precision score improved as seen in Table 1.

Metrics	YOLO	SSD	Classification
Recall	0.9512	0.7075	0.9355
Precision	0.806	0.6414	0.9063
mAP:@0.5	0.9533	0.9535	N/A
mAP:@0.95	0.6439	0.6412	N/A

Table 1: Precision, Recall, and mAP across different models.

4 Conclusion

In this paper, we created a dataset for a low-resource sign language and we experimented with different models and deployed the best model for real-time sign-to-speech synthesis. In pursuance of the objective of this paper which is to bridge the communication barrier between the hearing impaired community and the larger society, we enabled text-to-speech synthesis and deployed the YOLO model for real-time usage in production. This is being done by converting the sign texts (or labels) that are being returned by the model into an equivalent voice of words using Pyttsx3, a text-to-speech conversion library in Python, and deploying the model on DeepStack Server for real-time usage. The authors hope that this work sufficiently demonstrates how much the alienation of the hearing impaired community by the larger society in developing countries can be mitigated using resource-efficient machine learning techniques and tools.

Acknowledgements

Many thanks to Amanda Bibire of Ogun State Broadcasting Corporation, for volunteering to create the first batch of the dataset. We appreciate the teachers and students of the Special Education School, Saint Peter's College, Abeokuta, for graciously dedicating 4 class sessions to create the later and larger batch of the dataset. Finally, we thank the ML Collective community for the generous computational support, as well as helpful discussions, ideas, and feedback on experiments.

References

[Everingham *et al.*, 2010] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[Li *et al.*, 2020] Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1459–1469, 2020.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[Liu *et al.*, 2016] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[Odartey *et al.*, 2019] Lamptey K Odartey, Yonfeng Huang, Effah E Asantewaa, and Promise R Agbedanu. Ghanaian sign language recognition using deep learning. In *Proceedings of the 2019 the International Conference on Pattern Recognition and Artificial Intelligence*, pages 81–86, 2019.

[Organization and others, 2018] World Health Organization et al. Addressing the rising prevalence of hearing loss. 2018.

[Redmon *et al.*, 2016] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[Sandler *et al.*, 2018] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.