Sign Language Recognition

CS 7641 Group 3

Ahmed Rabbani
Ahindrila Saha
Hechen Li
Jenna Gottschalk
Sai Yang

Problem Definition

Motivation

- According to the WHO, 5% of the world population suffers from some extent of speech - hearing impairment.
- We intend to use Sign language recognition, powered by machine learning translation to bridge the communication gap between the hearing and hearing impaired community.

Objective

- **Classify** the American Sign Language Images using supervised machine learning
- **Cluster** gesture images from different sign languages to compare character similarity

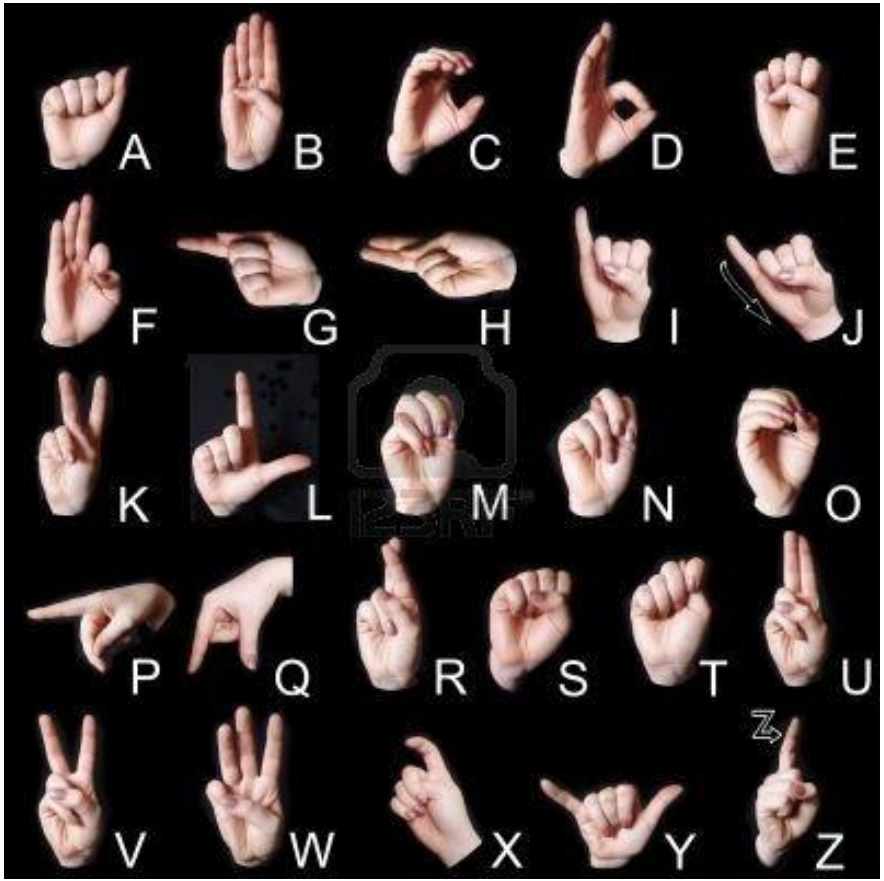
Literature Review

- In the past, PCA , Kurtosis position [1] and skeleton[4] have been used for feature extraction paired with CNN[3],Hidden Markov Chain[1] and SVM[2] for classification.
- Our approach requires only image input, supported by feature extraction from MediaPipe framework.

Data Overview

Images of American Sign Language will be used in our analysis. Images will be filtered for only 24 classes of English alphabets (A-Z) excluding J and Z due to rotation

Dataset	Number of features
Kaggle ASL Dataset	87,000 images of dimension 200*200 with 29 classes
ASL alphabet test dataset	870 images, 30 from each of the 29 classes



Data Preprocessing

Feature extraction

- Used **MediaPipe** to extract the 3D coordinates (x, y, z) of 21 key points on the hand for every image
- **63 features** extracted for each observation

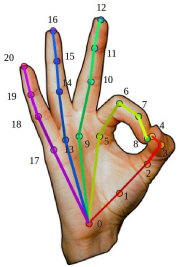
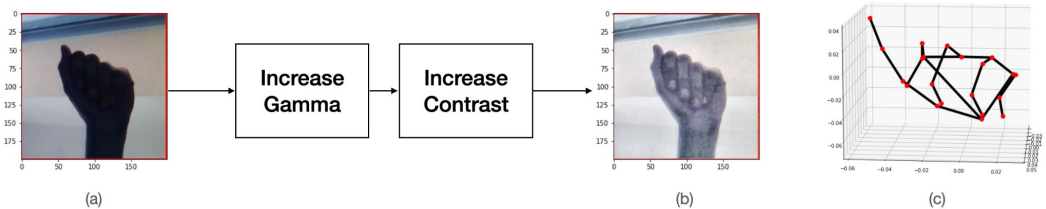


Image preprocessing

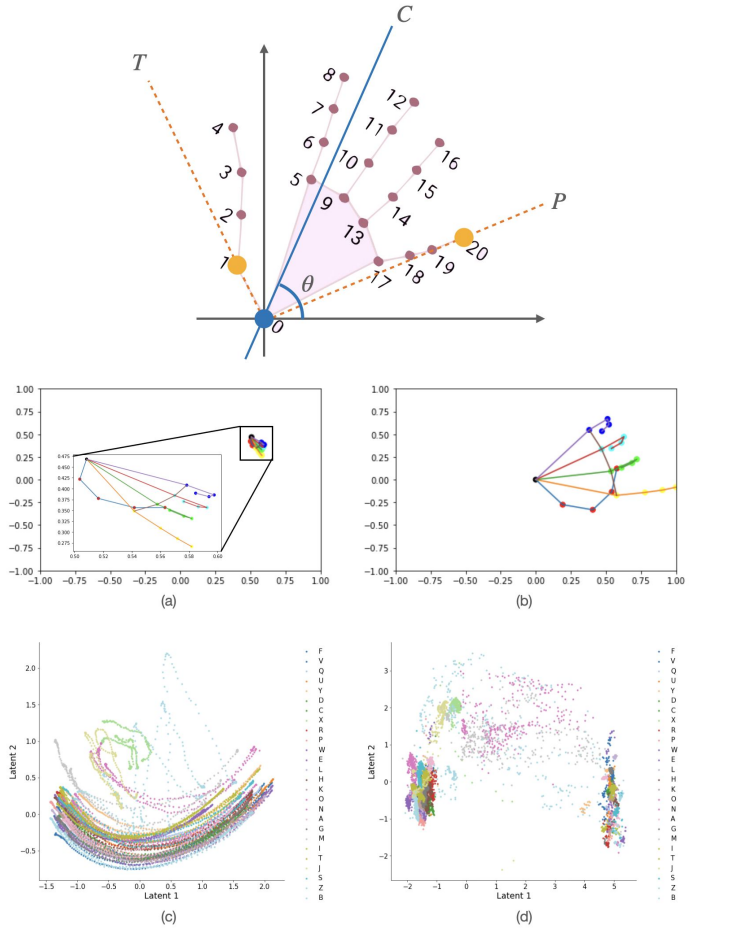
- Used **OpenCV** filters to **increase exposure** and **contrast** of dark images before applying MediaPipe
- Discarded character **“J”, “Z”, “Delete”, and “Space”** not processed properly by MediaPipe
- In total **59,822** preprocessed images (69% of the dataset) used for **supervised learning**



Data Preprocessing for Unsupervised Learning

Rotation Normalization

- Different rotation angles of images hinder unsupervised learning
- Normalizing the rotation of all images to the same scale and direction improves the clustering result



PCA before (c) and after (d) normalization

Methods

Supervised Classification

Feature engineering

- Dimensionality reduction (PCA / LDA)
- Feature selection (Lasso)

Model fitting

- **11** different ML models tested
- Hyperparameter tuning with cross-validation
- Model selection based on multiple performance metrics
- Retrained selected best model and tested its performance on hold-out data set

Unsupervised Clustering

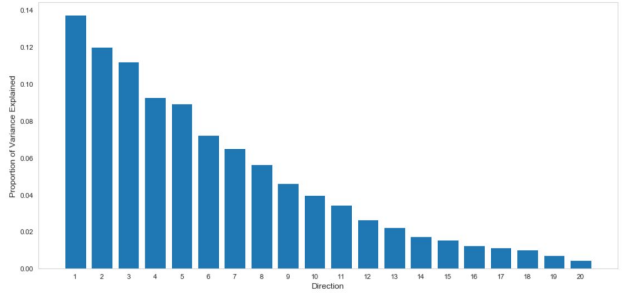
- Clustering on hand images from different sign languages
- Performance evaluation using both internal and external information

Supervised Classification

Feature Selection

- First 10 components of dimensionality reduction methods (PCA / LDA) explain over 95% variance
- Classification models perform the best with the **top 10 components selected by LDA**
- LDA selects such dimensions that the separability between different classes is maximized

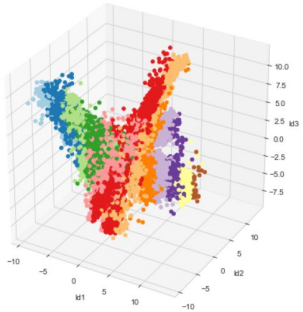
Distribution of variance explained from LDA components



Prediction accuracy

Techniques	KNN	SVM	Logistic Regression
PCA (10 components)	95.6%	86.5%	92.2%
LDA (10 components)	98.6%	92.7%	96.7%
LASSO	83.8%	67.2%	44.8%

Label clusters using first 3 LDA components



Model Training and Pre-selection

Overall model performance

- Precision, recall, F1-score as performance metrics
- Top 3 models:
  KNN, Random Forest, XGBoost
- Bottom 2 models:
  Logistic Regression and Naive Bayes

Label-wise model performance

- F1-score as performance metrics
- Top 2 models: **KNN, RF**
- Low classification accuracy for character M, N due to similarity

Accuracy Metrics	Bagging Classifier	Gradient Boosting Model1	KNN	Logistic Regression	Naive Bayes	Neural Networks	Random Forest	SVM Linear Kernel	SVM Polynomial Kernel	SVM Radial Kernel	XGBoost
Precision	97.2%	97.2%	98.5%	93.5%	93.8%	96.5%	98.3%	97.1%	96.6%	97.4%	98.0%
Recall	97.1%	97.2%	98.5%	93.2%	93.5%	96.5%	98.2%	97.1%	96.0%	97.3%	98.0%
F1 Score	97.1%	97.2%	98.5%	93.2%	93.6%	96.5%	98.2%	97.1%	96.1%	97.4%	98.0%

labels	Bagging Classifier	Random Forest	Gradient Boosting Model	XGBoost	KNN	Logistic Regression	Naive Bayes	Neural Networks	SVM Linear Kernel	SVM Polynomial Kernel	SVM Radial Kernel
A	97.9%	99.0%	97.2%	98.7%	98.4%	94.6%	95.9%	96.7%	98.6%	96.8%	98.0%
B	99.0%	99.4%	99.5%	99.3%	98.8%	97.4%	98.6%	99.3%	99.3%	98.4%	98.9%
C	99.2%	99.4%	98.5%	99.0%	99.4%	97.9%	97.8%	97.0%	98.7%	97.9%	98.6%
D	98.4%	99.1%	98.4%	99.1%	98.9%	96.5%	96.9%	97.8%	98.1%	98.5%	98.4%
E	98.3%	99.3%	98.8%	98.9%	98.8%	96.4%	97.1%	96.6%	98.5%	96.4%	98.3%
F	99.6%	99.6%	99.4%	99.5%	99.5%	99.1%	99.2%	99.1%	99.1%	99.6%	99.3%
G	98.2%	99.1%	98.8%	99.2%	99.7%	97.8%	98.4%	97.9%	98.9%	99.1%	98.8%
H	98.3%	99.1%	98.8%	99.0%	99.5%	97.4%	97.2%	97.8%	98.8%	98.6%	98.8%
I	97.9%	99.1%	98.0%	98.6%	99.2%	98.0%	97.5%	97.6%	99.0%	98.7%	99.1%
K	98.6%	99.6%	99.2%	99.3%	99.7%	97.0%	94.6%	98.8%	99.5%	99.4%	99.5%
L	99.6%	99.5%	99.3%	99.6%	99.5%	99.4%	99.0%	99.5%	99.5%	99.7%	99.6%
M	88.3%	92.2%	88.4%	91.1%	94.6%	82.6%	73.4%	86.6%	86.0%	87.1%	87.6%
N	86.0%	90.8%	88.1%	89.1%	93.7%	81.8%	77.4%	84.4%	84.1%	85.1%	87.5%
O	98.5%	99.0%	98.5%	99.0%	98.6%	96.0%	96.8%	97.5%	98.3%	98.0%	98.3%
P	98.0%	99.1%	98.2%	98.7%	98.9%	94.6%	93.8%	96.6%	97.2%	97.6%	97.7%
Q	98.4%	99.2%	97.7%	98.6%	98.8%	94.9%	94.5%	96.6%	97.5%	98.3%	98.2%
R	94.4%	96.0%	94.0%	96.5%	97.1%	75.2%	80.2%	94.0%	94.7%	90.2%	94.1%
S	97.2%	98.8%	97.7%	98.1%	98.6%	94.7%	96.8%	97.5%	98.0%	97.3%	98.0%
T	98.5%	98.8%	97.8%	98.8%	98.7%	97.5%	97.3%	97.4%	98.6%	98.5%	98.6%
U	94.2%	96.2%	94.7%	96.4%	97.4%	60.3%	79.0%	95.0%	93.9%	92.2%	94.6%
V	97.8%	98.8%	98.2%	98.9%	99.3%	94.4%	90.5%	98.6%	98.3%	98.5%	98.9%
W	97.9%	98.4%	97.8%	98.3%	98.5%	98.0%	98.7%	98.2%	98.0%	98.5%	98.8%
X	97.6%	98.1%	96.6%	97.6%	98.2%	95.5%	95.8%	96.3%	97.8%	82.9%	97.5%
Y	99.7%	99.7%	99.4%	99.7%	99.8%	99.2%	99.3%	99.0%	99.7%	99.4%	99.7%

Hyperparameter Tuning

Searching for optimal parameter values

- Pre-selected models: **KNN, Random Forest, XGBoost**
- Used **random search** in scikit-learn with cross-validation for hyperparameter tuning
- Optimal parameter values found
  - XGBoost: {'subsample': 0.6, 'n_estimators': 300, 'min_child_weight': 5, 'max_depth': 7, 'learning_rate': 0.1, 'gamma': 1.5, 'colsample_bytree': 0.8, 'alpha': 0}
  - KNN: {'p': 2(euclidean_distance), 'n_neighbors': 2, 'leaf_size': 30}
  - Random Forest: {'n_estimators': 800, 'max_features': 'sqrt'}

Final model selection

- **KNN** performs best on the validation set among three models
- Best performance on both overall and label-wise accuracy metrics

Accuracy Metrics	KNN	XGBoost	Random Forest
F-1 score	99.0%	98.0%	98.0%
Precision	99.0%	98.0%	98.0%
Recall	99.0%	98.0%	98.0%

Labels	KNN	Xg Boost	Random Forest
A	99.0%	99.0%	99.0%
B	99.0%	100.0%	99.0%
C	99.0%	99.0%	99.0%
D	99.0%	98.0%	99.0%
E	99.0%	99.0%	99.0%
F	99.0%	100.0%	100.0%
G	100.0%	99.0%	99.0%
H	99.0%	99.0%	99.0%
I	99.0%	99.0%	99.0%
K	100.0%	100.0%	100.0%
L	100.0%	100.0%	100.0%
M	93.0%	93.0%	93.0%
N	91.0%	89.0%	91.0%
O	99.0%	98.0%	99.0%
P	99.0%	99.0%	99.0%
Q	99.0%	98.0%	99.0%
R	98.0%	96.0%	96.0%
S	99.0%	98.0%	99.0%
T	99.0%	98.0%	99.0%
U	98.0%	96.0%	97.0%
V	99.0%	98.0%	99.0%
W	99.0%	98.0%	99.0%
X	98.0%	98.0%	98.0%
Y	100.0%	100.0%	100.0%

Result Evaluation

- Hold-out test set of images with extremely dark and noisy background
- Lower prediction performance than cross-validation result
- Improvement suggestion: Convoluted Neural Network

Prediction accuracy of KNN on hold-out test set

F-1 score	Precision	Recall
84.0%	86.0%	85.0%

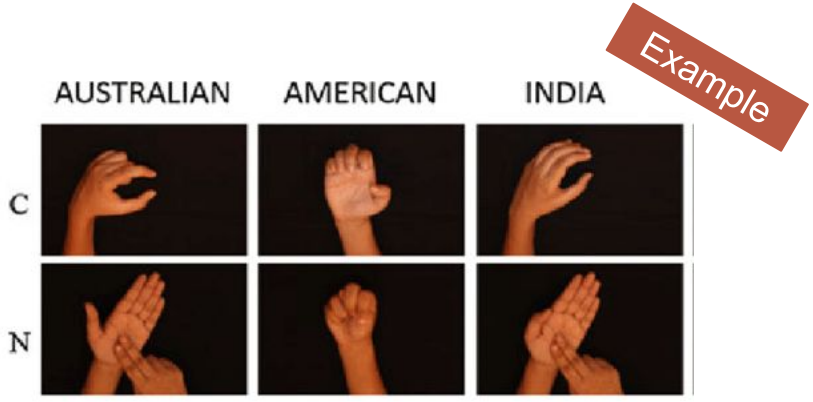
Labels	A	B	C	D	E	F	G	H	I	K	L	M	N
F-Score	90.0%	100.0%	78.0%	88.0%	95.0%	100.0%	98.0%	100.0%	98.0%	84.0%	91.0%	58.0%	21.0%

Labels	O	P	Q	R	S	T	U	V	W	X	Y
F-Score	86.0%	100.0%	100.0%	64.0%	94.0%	83.0%	38.0%	64.0%	100.0%	80.0%	100.0%

Unsupervised Clustering

Unsupervised Clustering

- Objective: Find similar gestures across different sign languages
- Input: **124,044** preprocessed images of **104** unique characters from **4** different sign languages
- Output: clusters of similar sign language gestures
- Method: K-means

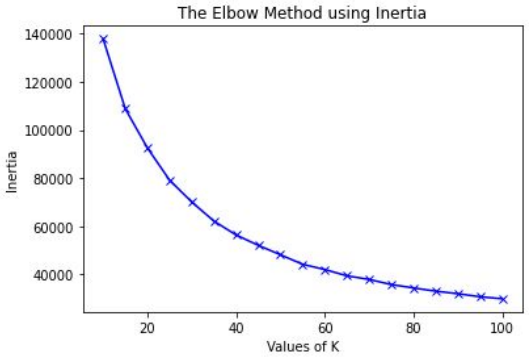


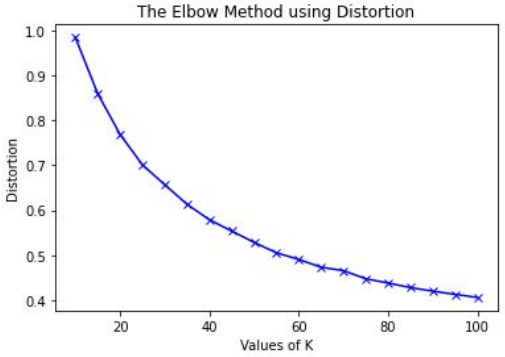
Data Input

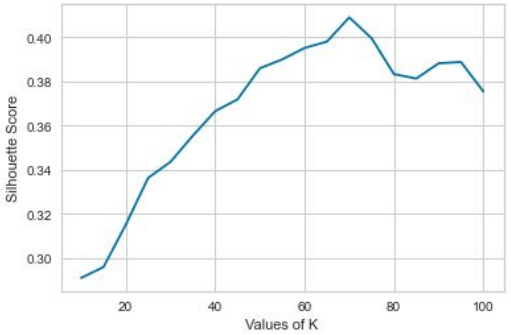
Sign Language	Number of Images	% Share
American	69,254	56%
Indian	41,891	34%
Irish	8,850	7%
Brazilian	4,049	3%

Optimal Number of Clusters

- Test range of K: 10 - 100
- Performance measure:
  Elbow method & Silhouette score
- Highest Silhouette score 0.41 at K = 70
- Images from the same characters and languages separated into more clusters as K increases, but without significant increase in performance
- Selected K = 30 for better interpretability of clustering result

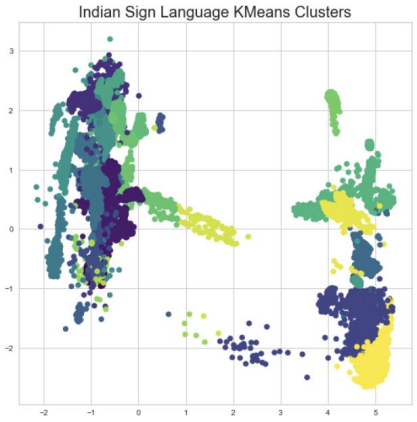


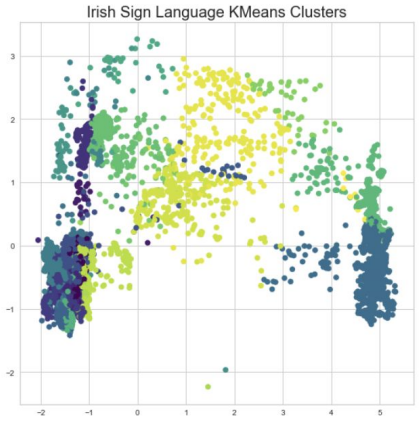


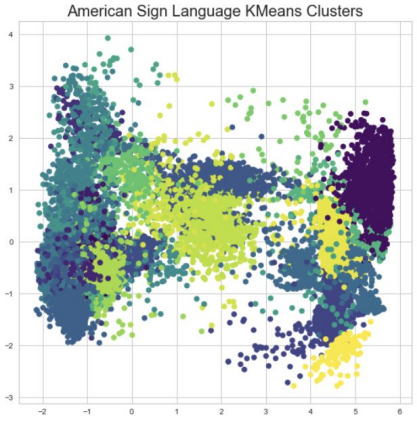


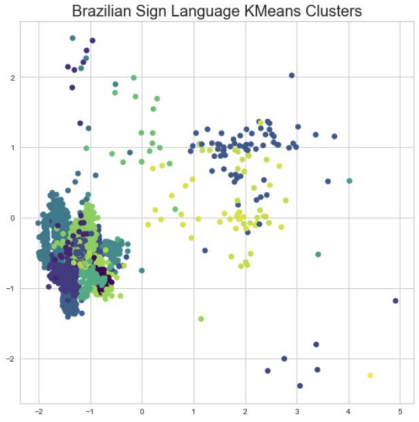
Clustering Result

- Cluster distribution of ASL and Irish Sign Language less distinguishable in the first two principal components
- More clear boundaries between individual clusters of Indian Sign Language
- Gestures in the Indian Sign Language more distinct from each other than in the ASL and Irish Sign Language









Matching-based Performance

- Performance measure based on external information - ground truth labels: characters and languages
- Average F-1 score: 0.54
- Dominating labels in the best performing clusters:
  {9, Indian}, {W, American},
  {F, American}, {D, American}

Cluster	Character Label of maximum number of data points	Language Label of maximum number of data points	precision	recall	F-1
16	9	Indian	0.915267176	0.999166667	0.955378486
18	W	American	0.765977444	0.961462839	0.852659111
3	F	American	0.739949109	0.990125979	0.846949177
23	D	American	0.813785637	0.834875972	0.824195906
29	O	Indian	0.594908688	0.895833333	0.714998337
14	L	American	0.490579978	0.992526158	0.656613103
25	Z	American	0.442404945	0.946693387	0.60301251
15	U	Indian	0.416968818	0.958333333	0.581101566
4	6	Indian	0.40417649	1	0.575677621