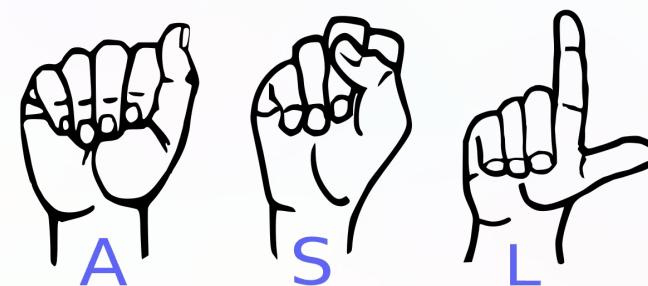


# Unlocking Communication:

## Sign Language to Text Conversion



**For Dumb and Deaf**

# Project Abstract: Bridging Communication Gaps

Our mission is to empower the deaf and hard-of-hearing community by leveraging cutting-edge technology to facilitate seamless communication. Our innovative project aims to:

→  **Develop a Computer Application**

Crafting a user-friendly software solution designed for accessibility and real-time interaction.

→  **Train an Advanced AI Model**

Utilizing machine learning to accurately interpret diverse hand gestures from American Sign Language (ASL).

→  **Real-time Video Interpretation**

Processing live video feeds of ASL gestures to instantly translate them into on-screen text.

→  **Enable Instant Text Output**

Providing immediate textual conversion of signs, enhancing communication fluidity and understanding.

# The Core Components of Sign Language:

Sign language is a rich and expressive visual language, built upon a foundation of key elements that convey meaning. Understanding these components is crucial to mastering and appreciating its intricacies.

## **Handshape**

The specific configuration of the hand used to form a sign. Think of it as the "alphabet" of sign language, where each shape represents a distinct unit of meaning.

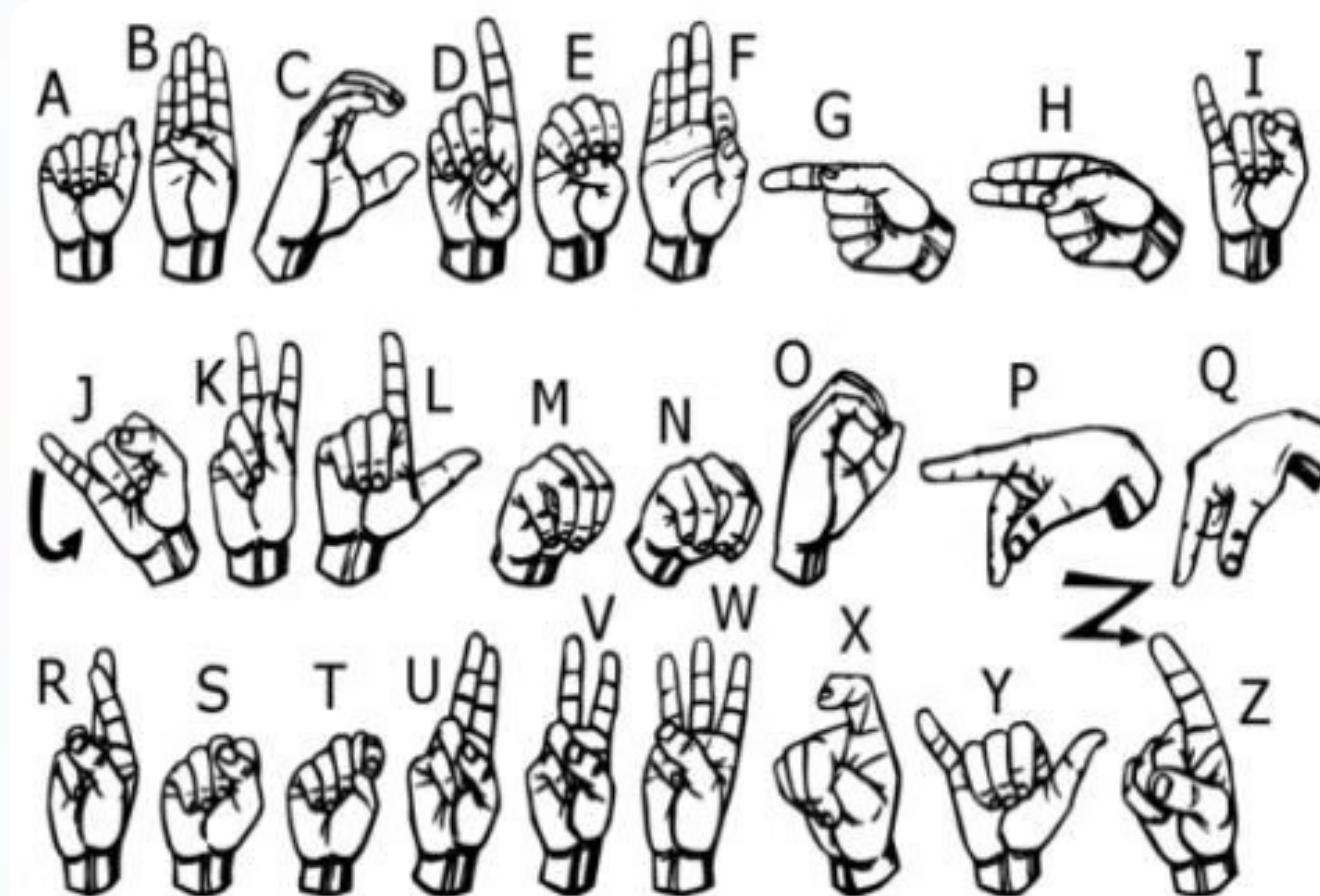
## **Movement**

The path, direction, and repetition of the hand(s) in space. This dynamic element adds nuance and grammatical information, transforming static handshapes into expressive actions.

## **Non-Manual Features**

Beyond the hands, these include facial expressions, body posture, and head movements. They are essential for conveying emotion, grammatical information, and contextual meaning, akin to tone of voice in spoken language.

# Mastering the ASL Alphabet: 27 Key Symbols!



Our project proudly recognizes 27 **essential American Sign Language symbols**, encompassing the entire alphabet from A to Z, plus the crucial 'blank' gesture. This comprehensive set forms the foundation of our innovative ASL recognition system.

# Unveiling Our Data Journey



## Phase 1: Data Acquisition & Discovery 🏗️

We embarked on a meticulous search to gather diverse and relevant raw data from various sources, laying the foundational bricks for our analysis.



## Phase 2: Intelligent Data Annotation 🎯

Our raw data underwent precise labeling and structuring, transforming unstructured information into a format ready for AI consumption and learning.



## Phase 3: Rigorous Data Cleansing 💧

A critical step to eliminate noise, inconsistencies, and missing values, ensuring our dataset was pristine and reliable for accurate model training.



## Phase 4: Feature Engineering & Transformation 🛡️

Here, we crafted powerful new features and applied various transformations to optimize the dataset, making it perfectly tailored for our machine learning models.



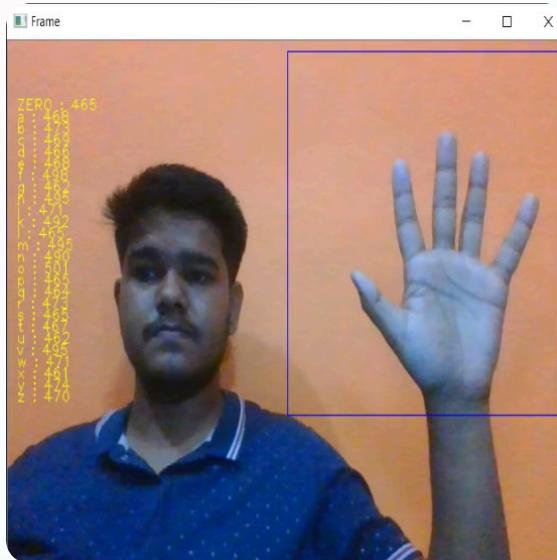
## Phase 5: Quality Assurance & Validation ✅

Every step was rigorously validated to guarantee data integrity, robustness, and readiness, empowering our models with high-quality, actionable insights.

# Data Generation & Preprocessing Journey

## Capturing Raw Images

Our journey begins with capturing the initial, unprocessed visual data. These raw images are the foundational input, rich in detail and directly representing the real-world scenarios our model will learn from.



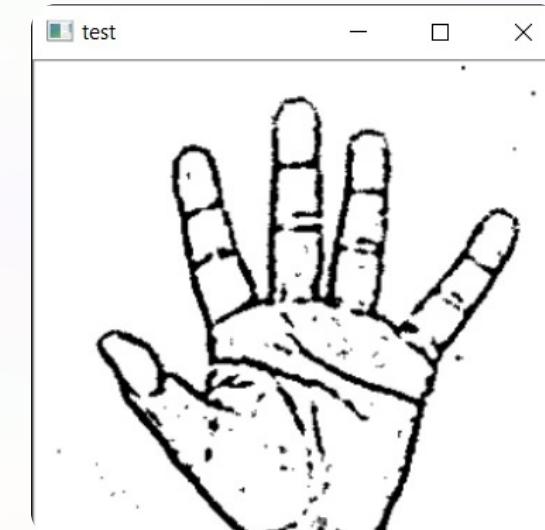
## Grayscale Transformation

To simplify complexity and focus on essential structural information, we convert the colorful raw images into grayscale. This crucial step reduces computational load while preserving key features necessary for robust analysis.



## Gaussian Blur Application

Next, we apply a Gaussian blur to the grayscale images. This technique effectively reduces image noise and smooths out fine details, making the main features more prominent and improving the stability and generalization of our models.



## Why We Forged Our Own Path: The Power of a Custom Dataset

### Unmatched Relevance

Existing datasets often lack the specific nuances and diversity required for our unique problem. Building our own allowed us to capture data perfectly aligned with our research goals.

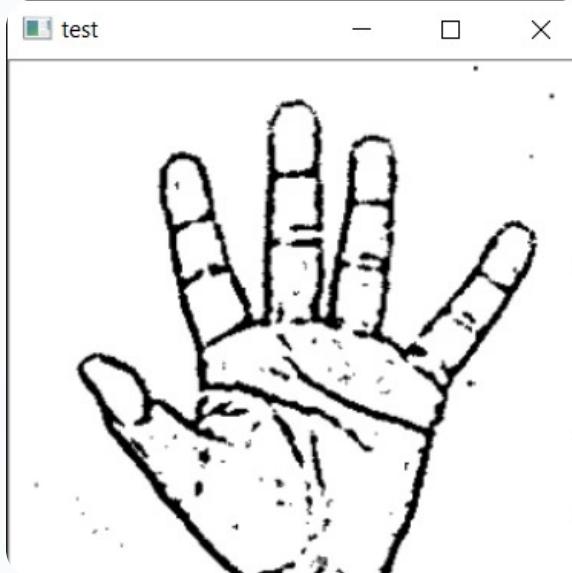
### Superior Data Quality & Control

By curating our dataset from scratch, we ensured high-fidelity annotations, consistent labeling, and robust data integrity, minimizing noise and maximizing accuracy for training our models.

### Tackling Novel Challenges

For groundbreaking work, off-the-shelf solutions aren't enough. Our custom dataset enabled us to address previously unexplored aspects and innovate beyond current limitations.

# Gesture Classification: Bridging Communication with AI



## Layer 1: Precision Classification 🔎

Unlocking the ability to **Classify between Similar Symbols** by discerning subtle nuances and fine details within complex gestures. This layer handles the intricate variations that demand high accuracy.



## Layer 2: Comprehensive Recognition 🧠

Expanding the scope to **Classify between 27 Distinct Symbols**, enabling broad and robust gesture recognition across a diverse set of commands and expressions.

# Algorithm Layer 1

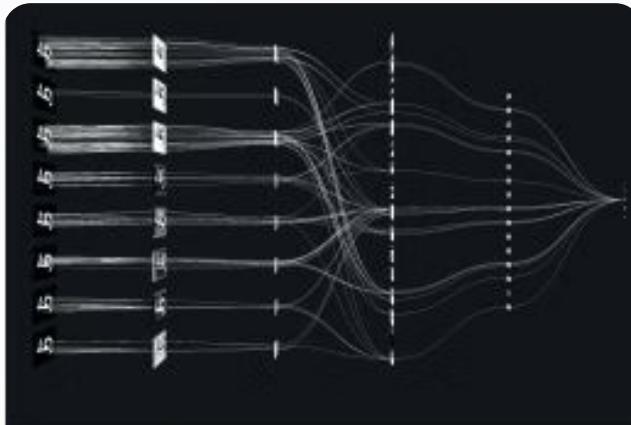
This initial layer is all about transforming raw visual input into meaningful data, preparing it for intelligent interpretation.

-  **Image Pre-processing:** We kick things off by applying a Gaussian blur filter and a dynamic threshold to raw video frames using OpenCV. This refines the image, making crucial features pop for optimal extraction.
-  **CNN Prediction Power:** The pre-processed image then fuels our Convolutional Neural Network (CNN). For a letter to be confirmed and recorded, it must be consistently detected across more than 50 consecutive frames – ensuring robust and accurate predictions for word formation.
-  **Seamless Word Formation:** To build complete words, we intelligently incorporate "blank" symbols. These clever markers denote the spaces between detected signs, allowing us to construct coherent and correctly spaced words from the continuous stream of hand gestures.

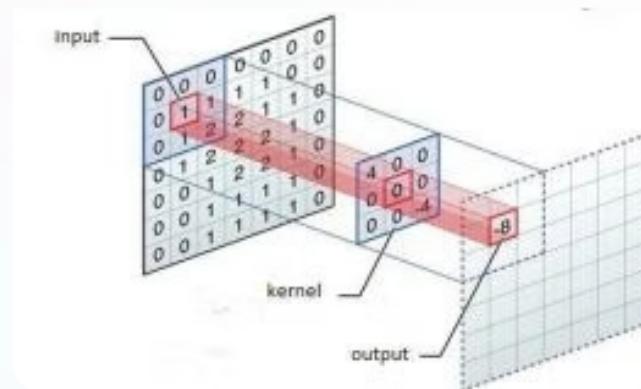
## Algorithm Layer 2

- **💡 Unveiling Ambiguities:** We pinpoint various symbol sets that exhibit similar detection patterns, posing unique challenges for accurate recognition.
- **🎯 Precision Classification:** Dedicated classifiers are then employed, meticulously crafted to differentiate between these challenging symbol groups.
- **⚠️ Decoding Challenges:** During rigorous testing, certain symbols presented significant detection ambiguities, often being mistaken for others:
  - **D** was often confused with: R and U
  - **U** was often confused with: D and R
  - **I** was often confused with: T, D, K, and I (itself, indicating broad ambiguity)
  - **S** was often confused with: M and N

# Convolutional Neural Networks (CNNs)



- The Learning Journey: Starting with random configurations, these "filters" evolve through rigorous training, progressively honing their feature extraction capabilities.
- Core Application: At their heart, CNNs are masterful at image classification, discerning patterns and categories with remarkable precision.
- Unveiling the Architecture: CNNs are built upon layers of convolutional power, each packed with "filters" designed for intricate feature extraction.



## Our CNN Classifier Model

Dive into the core of our solution: a powerful Convolutional Neural Network meticulously designed for cutting-edge image classification. This model is engineered to distinguish patterns with remarkable accuracy and efficiency.

1

### Robust Architecture

Our model features a multi-layered convolutional design, incorporating advanced activation functions and pooling strategies. This ensures effective feature extraction and robust pattern recognition across diverse datasets, building on the foundational principles of CNNs.

2

### Precision Training

Trained on extensive and carefully curated datasets, our CNN benefits from rigorous optimization techniques. This comprehensive training regimen hones its ability to generalize, making it highly effective even with unseen data.

3

### High-Performance Output

Achieving superior accuracy and minimal latency, our classifier model delivers reliable results for critical applications. Its optimized structure ensures both high predictive power and efficient resource utilization.

4

### Innovative Edge

Beyond standard configurations, our model integrates novel architectural enhancements tailored to specific challenges, pushing the boundaries of what's possible in automated image analysis. This allows for adaptability and scalability.

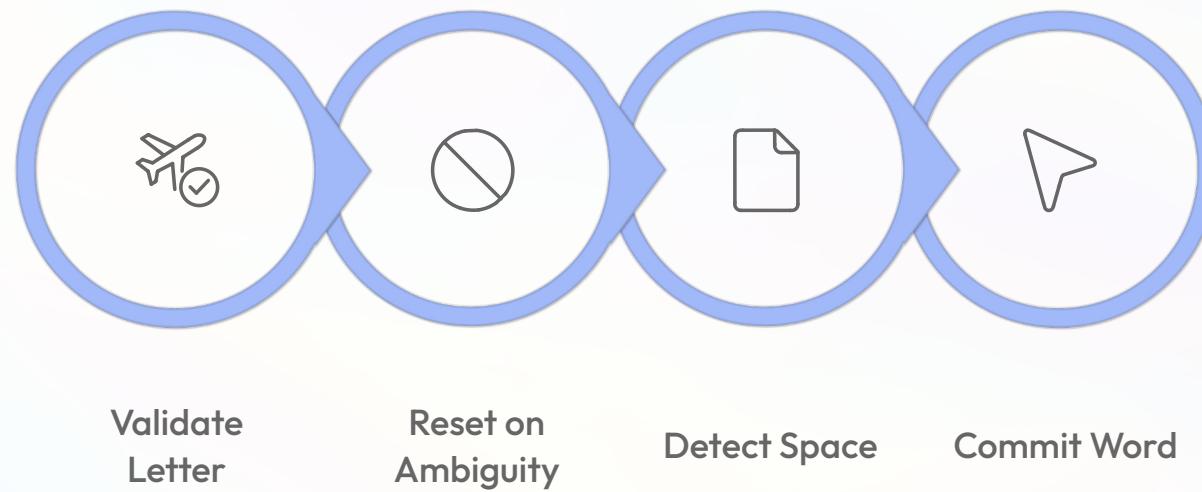
## Power of Our CNN Classifier Model

Dive deep into the core of our solution: a cutting-edge Convolutional Neural Network (CNN) Classifier designed to revolutionize how we process and understand visual data. This model stands at the forefront of AI innovation, delivering unparalleled accuracy and efficiency in complex classification tasks.

-  **State-of-the-Art Architecture:** Built with advanced layers and optimized for high-performance computing, ensuring rapid and precise classifications.
-  **Exceptional Accuracy:** Achieves superior recognition rates across diverse datasets, minimizing errors and maximizing reliability.
-  **Scalable & Adaptable:** Engineered for flexibility, our CNN can be readily scaled and fine-tuned for a wide range of applications and data types.
-  **Robust Feature Extraction:** Automatically learns and identifies intricate patterns and hierarchies within data, providing deep insights without manual engineering.

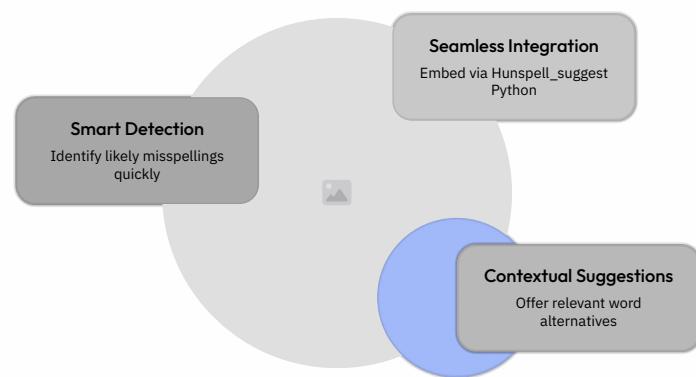
# System Implementation: From Gesture to Text!

1. ✨ **Confident Letter Validation:** When a single letter's detection count confidently surpasses a predefined threshold (e.g., 50 detections in our code) and maintains a clear distinction from other potential letters (difference threshold of 20), it's recognized and appended to the current word.
2. ❌ **Preventing Misinterpretation:** To ensure accuracy and avoid erroneous predictions, if detections are ambiguous or lack clear dominance, the current symbol detection counts are reset, awaiting a more definitive input.
3. 🧑 **Smart Space Recognition:** A blank (plain background) detection exceeding a specific value signifies a pause. However, if the current word buffer is empty, this pause is ignored to prevent unnecessary spaces.
4. ✎ **Word Completion & Sentence Formation:** When a significant blank is detected after a formed word, it triggers the insertion of a space, marking the end of a word, and seamlessly appends it to the evolving sentence.



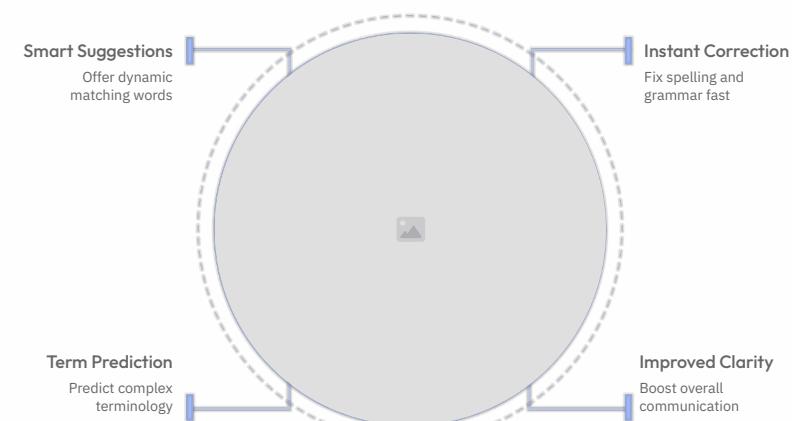
# Smart Autocorrect: Enhancing Accuracy

## Intelligent Word Suggestions 🧠



Harnessing the power of the **Hunspell\_suggest** Python library, our system proactively analyzes your input, providing intelligent suggestions for potentially incorrect words. This ensures your message is always clear and precise.

## Seamless Correction & Prediction ✨



Users are empowered to effortlessly choose from a dynamic set of matching words, instantly refining their sentences. This feature not only dramatically reduces spelling mistakes but also brilliantly assists in predicting complex terminology, boosting communication efficiency.

# Overcoming Hurdles: Our ASL Recognition Journey

1

**Feature Extraction Filter Selection** : The quest for optimal feature extraction involved extensive experimentation. After exploring various filters like binary threshold and Canny edge detection, Gaussian blur proved to yield superior results, becoming our chosen method.

2

**Model Accuracy Enhancement** : Initial phases presented challenges with model accuracy. We systematically addressed this by increasing input image size and continually refining our dataset, leading to significant improvements in our model's performance.

3

**Dataset Creation Challenge** : Unable to find a comprehensive dataset of raw ASL character images, we took the initiative to meticulously build our own, ensuring a robust foundation for our model.

# Essential Development Tools

 Python 3.6.6

 Tensorflow 1.11.0

 OpenCV 3.4.3.18

 NumPy 1.15.3

 Matplotlib 3.0.0

 Hunspell 2.0.2

 Keras 2.2.1

 PIL 5.3.0

# Navigating Model Limitations

-  **Lighting Dependency:** Our model currently delivers optimal performance predominantly in environments with *excellent lighting conditions*, where visual clarity is paramount.
-  **Background Sensitivity:** For achieving the highest detection accuracy, the model requires a *plain and uncluttered background*, minimizing visual noise and distractions.

## Conclusion:

1

👉 Developed a pioneering, real-time vision-based American Sign Language (ASL) recognition system specifically designed for the Deaf & Mute community, focusing on ASL alphabets.

2

✨ Achieved an exceptional accuracy rate of **98.00%** on our comprehensive dataset.

3

🧠 Significantly improved prediction capabilities by implementing a sophisticated two-layer algorithmic framework, enhancing verification and discrimination of highly similar ASL symbols.

## Future Horizons: Expanding Our Vision

-  **Conquering Complex Backgrounds:** Our next step involves exploring advanced background subtraction algorithms to achieve even higher accuracy in challenging, dynamic environments.
-  **Mastering Low-Light Conditions:** We aim to enhance our preprocessing techniques to ensure robust and accurate gesture prediction even in dimly lit or adverse lighting scenarios.

# Our Project Innovators Team



Anukalp Rai

220182



Anwesha Dwivedi

220190



Bhavya Sharma

220



Ishan Mukundhan

220



Norah Srivastava

220736

# Thank You!