

Klasifikasi Indonesian Sign Language menggunakan EfficientNet

Kelompok 4

C14180006 - Tomy Widjaja

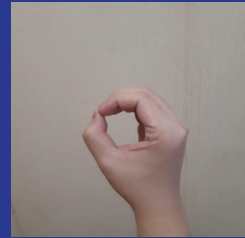
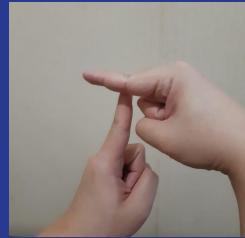
C14180013 - Enrico Hartanto

C14180014 - Henry Wicaksono

C14180029 - Andrew Firman

C14180046 - Gregorius Nicholas

C14180057 - Gregorius Kevin



Introduction

Sign language merupakan bahasa yang digunakan untuk berkomunikasi dengan orang yang tuna rungu. Untuk berkomunikasi menggunakan sign language, diperlukan keterampilan dalam mengartikan sign language tersebut. Oleh karena itu, diperlukan sistem yang dapat mengklasifikasikan gambar bentuk tangan untuk Indonesian Sign Language.



Related Works

Sign Language Interpreter using Deep Learning (Gupta et al., 2019)

- **Metode:** Deep Learning, CNN Model.
- **Dataset:** American Sign Language gestures.
- **Hasil:** Mampu mengklasifikasi sign language dengan akurasi >96%.



Related Works

Pengenalan Bahasa Isyarat Indonesia (BISINDO) Untuk Karakter Huruf Dengan Menggunakan Microsoft Kinect (Yunanda et al., 2018)

- **Metode:** Hidden Markov Model (HMM)
- **Dataset:** Gerakan tangan yang diambil dengan menggunakan Microsoft Kinect meliputi:
 - ❖ Huruf: a, i, u, e, o.
 - ❖ Kata: berita, kunjung, rambut
- **Hasil:** Uji coba dilakukan dengan pengambilan gerakan tangan dengan jarak 80 cm dan 150 cm dari sensor. Hasilnya adalah akurasi antara 75% untuk jarak 150 cm dan 87,5% untuk jarak 80 cm.

Dataset

Sumber Data:

- Dataset BISINDO
URL: <https://www.kaggle.com/idhamozi/indonesian-sign-language-bisindo>.
- Dataset hasil foto sendiri



Dataset

Dataset Pengujian 1

- **Training Data**
 - ❖ 64 foto tangan dari dataset BISINDO
 - ❖ **Total:** 1728 foto (64 foto per kelas)
- **Testing Data**
 - ❖ 2 foto tangan Henry per kelas
 - ❖ **Total:** 54 foto (2 foto per kelas)



Dataset

Dataset Pengujian 2

- **Training Data**

- ❖ 10 foto tangan Andrew per kelas
- ❖ 10 foto tangan Enrico per kelas
- ❖ 10 foto tangan Gregorius per kelas
- ❖ 10 foto tangan Henry per kelas
- ❖ 10 foto tangan Nicholas per kelas
- ❖ 10 foto tangan dari dataset BISINDO per kelas
- ❖ **Total:** 1620 foto (60 foto per kelas)



Dataset

Dataset Pengujian 2 (cont.)

- **Seen Testing Data** (Tangan terdapat pada training data)
 - ❖ 2 foto tangan Andrew per kelas
 - ❖ 2 foto tangan Enrico per kelas
 - ❖ 2 foto tangan Gregorius per kelas
 - ❖ 2 foto tangan Henry per kelas
 - ❖ 2 foto tangan Nicholas per kelas
 - ❖ 2 foto tangan dari dataset BISINDO per kelas
 - ❖ **Total:** 324 foto (12 foto per kelas)



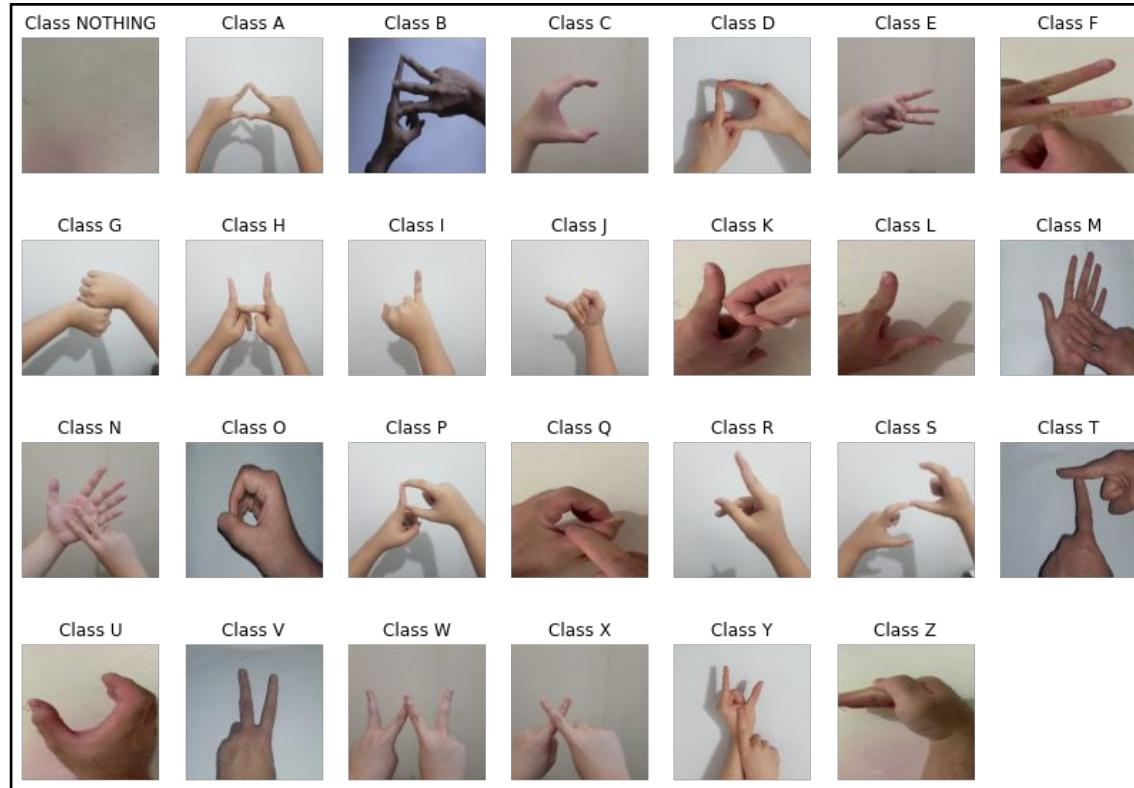
Dataset

Dataset Pengujian 2 (cont.)

- **Unseen Testing Data** (Tangan tidak terdapat pada training data)
 - ❖ 15 foto tangan Tomy per kelas
 - ❖ **Total:** 405 foto (15 foto per kelas)



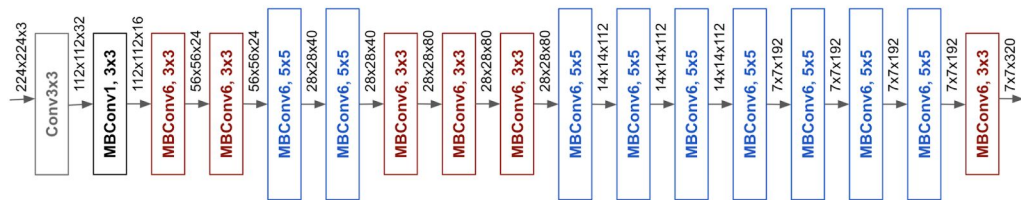
Contoh Dataset Sendiri



Method

Implementasi dilakukan dalam bahasa pemrograman **Python** dengan menggunakan library **Tensorflow** dan **Keras**.

Proses transfer learning menggunakan **EfficientNet B0** dengan parameter di samping:



- Loss function: **Categorical Cross-Entropy**
- Optimizer: **Rectified Adam**
 - Learning rate = 1e-3
 - min_lr = 1e-7
 - warmup_proportion = 0.15
- Callback:
 - **Learning Rate Reducer**
 - monitor = 'val_accuracy'
 - patience = 5
 - verbose = 1
 - factor = 0.2
 - min_lr = 1e-7
 - **Early Stopping**
 - monitor = 'val_accuracy'
 - mode = 'max'
 - patience = 5
 - restore_best_weights = True

Experiments

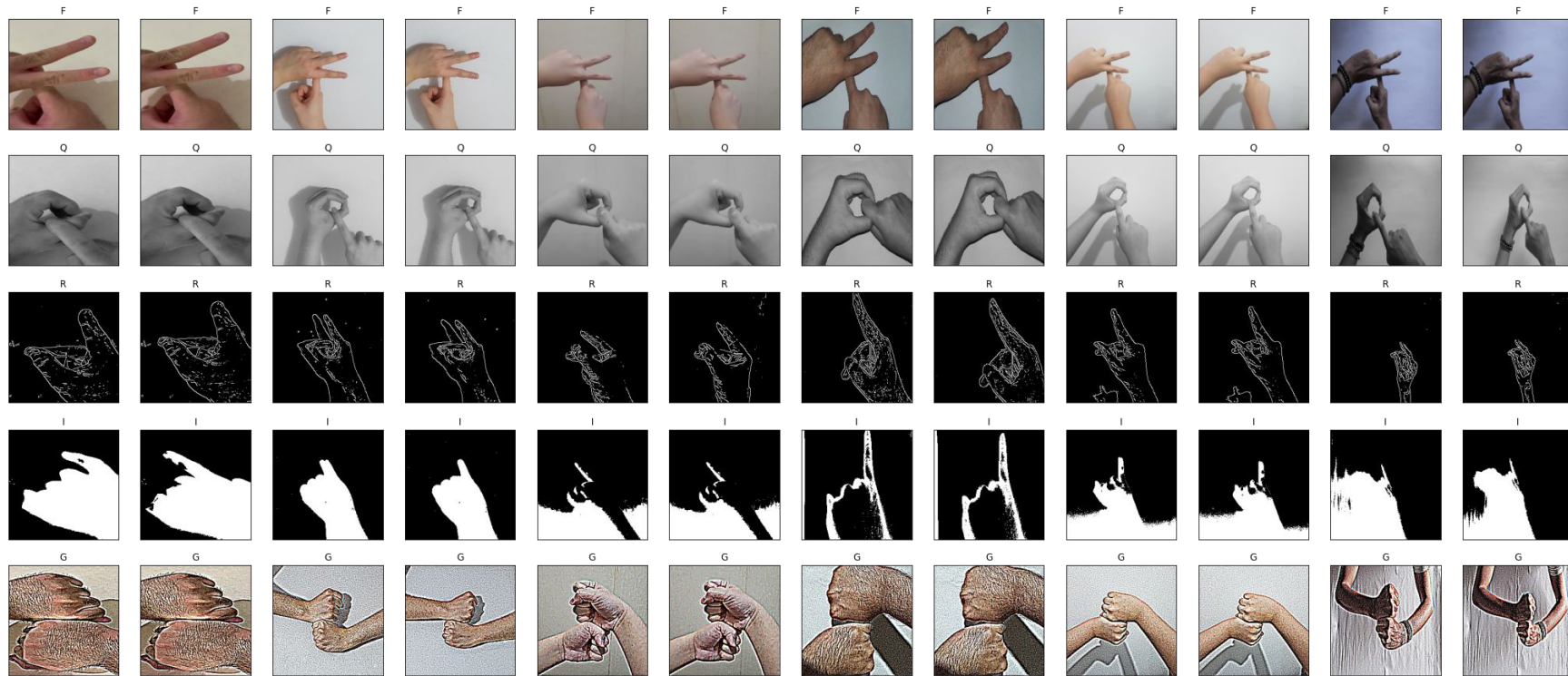
Seluruh image yang digunakan melalui proses resize terlebih dahulu untuk mengubah dimensinya menjadi 224x224x3.

Pengujian juga dilakukan dengan beberapa metode preprocessing yang berbeda:

- Raw (Tanpa preprocessing)
- Grayscale
- Edge Detection
- Segmentation
- Sharpening




Experiments



Experiments

Pengujian pertama menggunakan dataset BISINDO yang terdiri atas 64 image per kelasnya sebagai training data. Pengujian menggunakan hasil foto sendiri yang terdiri atas 2 image per kelasnya.

Setelah melalui berbagai tahapan preprocessing dan percobaan, didapati **hasil akurasi model tidak memuaskan pada pengujian pertama**. Hal ini diperkirakan disebabkan oleh dataset training yang hanya terdiri dari tangan satu orang. Model pun menjadi **overfit** dan memiliki akurasi sangat buruk saat digunakan untuk mengklasifikasikan hasil foto buatan sendiri.



Experiments

Pengujian kedua dilakukan dengan **menggunakan dataset buatan sendiri** (selanjutnya disebut sebagai Dataset Sendiri). Ditemukan bahwa konfigurasi **unfreeze top 20 layers pada EfficientNet B0 menghasilkan akurasi yang lebih baik** daripada konfigurasi unfreeze all layers.

Konfigurasi unfreeze top 20 layers digunakan dalam pengujian perbandingan berbagai metode preprocessing pada pengujian kedua: Raw, Grayscale, Edge Detection, Segmentation, Sharpening



Results

Pengujian 1 yang hanya menggunakan dataset BISINDO untuk training menghasilkan performa yang kurang baik pada hasil foto sendiri seperti pada tabel di bawah ini:

Precision	Recall	F1-Score
0.35	0.35	0.35

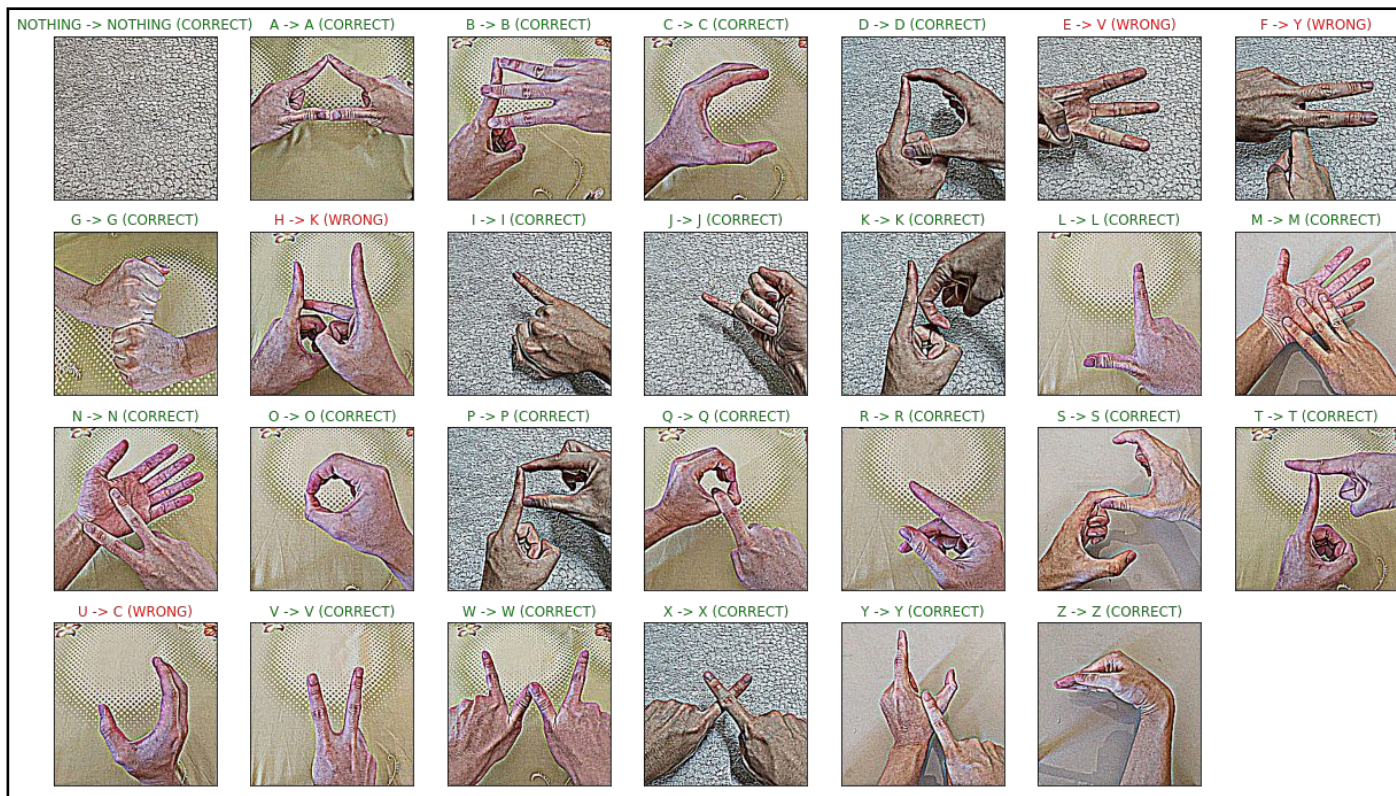
- **Masalah:** Training data kurang bervariasi
- **Solusi:** Pembuatan dataset yang lebih bervariasi → **Dataset Sendiri**



Results

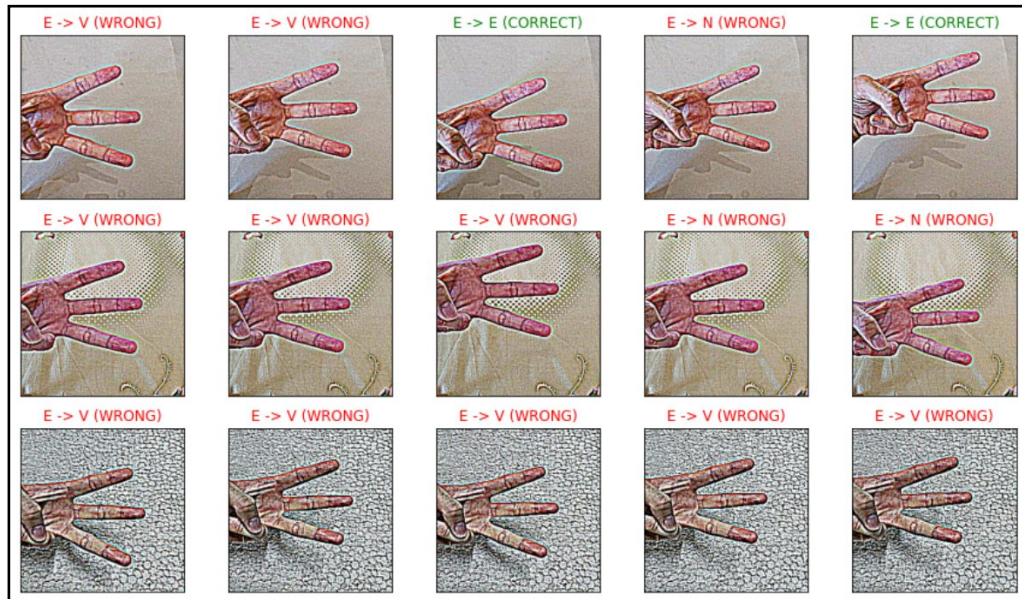
Preprocessing	Seen Test Data			Unseen Test Data		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Raw	0.91	0.91	0.91	0.81	0.81	0.81
Grayscale	0.9	0.9	0.9	0.7	0.7	0.7
Edge Detection	0.27	0.27	0.27	0.12	0.12	0.12
Segmentation	0.84	0.84	0.84	0.49	0.49	0.49
Sharpening	0.94	0.94	0.94	0.83	0.83	0.83

Results



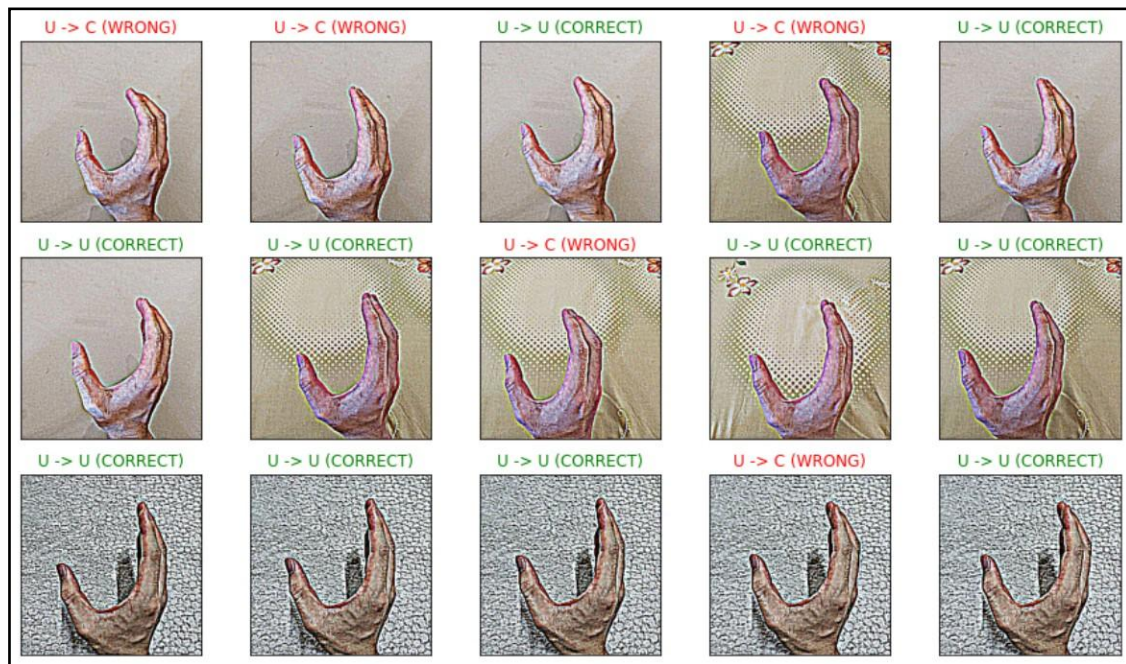
Conclusions

- Dataset yang bervariasi akan menghindarkan model dari *overfitting*
- Sharpening menghasilkan precision, recall, dan f1-score yang paling tinggi
- Setiap huruf rata-rata dapat terdeteksi dengan baik kecuali huruf E dan F



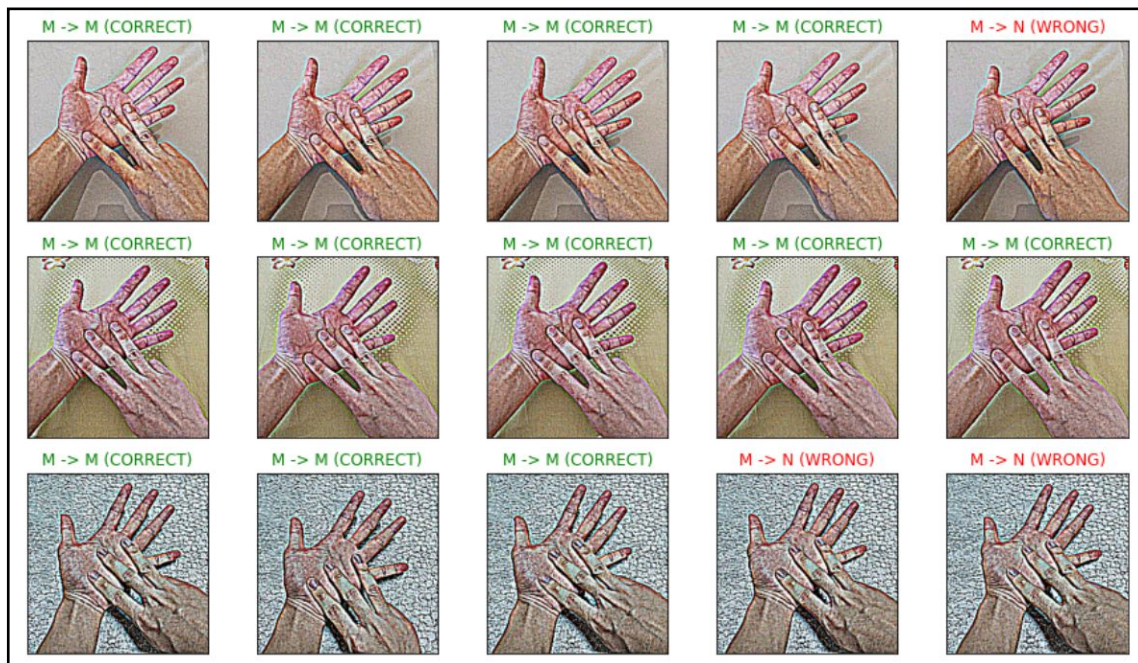
Conclusions

- Huruf C dan U terkadang sulit dibedakan karena klasifikasinya bersifat rotational invariance (hand sign U terlihat seperti hand sign C yang dirotasi)



Conclusions

- Huruf yang memiliki struktur yang hampir sama seperti huruf M dan N terkadang sulit dibedakan



References

- Yunanda, A. B., et al. (2018). Pengenalan Bahasa Isyarat Indonesia (BISINDO) Untuk Karakter Huruf Dengan Menggunakan Microsoft Kinect. Fountain of Informatics Journal, 3(2), 41-45. <http://dx.doi.org/10.21111/fij.v3i2.2469>.
- Gupta, H., et al. (2019) . Sign Language Interpreter using Deep Learning. Retrieved from <https://github.com/harshbg/Sign-Language-Interpreter-using-Deep-Learning>.

