Gesture Detection

▼ Resources

https://gurjeet333.medium.com/7-best-techniques-to-improve-the-accuracy-of-cnn-w-o-overfitting-6db06467182f

https://www.deeplearningbook.org/contents/guidelines.html

https://medium.com/video-classification-using-keras-and-tensorflow/action-recognition-and-video-classification-using-keras-and-tensorflow-56badcbe5f77

https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/

https://stackoverflow.com/questions/60889486/good-training-accuracy-but-poor-validation-accuracy

https://www.researchgate.net/publication/334003548_Dynamic_Hand_Gesture_Recognition_Based_on_3D_Convolutional

https://medium.com/@abdalrahman_shahrour/what-is-the-difference-between-cnn-rnn-and-lstm-ee24c7f3b3fd#:~:text=LSTM

Convolutional Neural Network

💡 Arhitectura aleasa pentru task-ul de hand gesture recognition este reprezentata de un 3DCNN (convolutional neural network 3D) + 2 layere LSTM (Long Short Term Memory)

💡 Motivatia pentru aceasta alegere e natura task-ului pe care il avem de indeplinit (ce inseamna un gest ?)

Ce inseamna un gest?

Un gest, in context-ul gesture detection, este o serie de frame-uri (input) care are nevoie de labeling. In cazul acesta, pentru a imbina secvente de mai multe frame-uri, avem nevoie de 2 componente:

- 3DCNN → procesare imagini, care are in plus axa temporala (o axa width, o axa height, o axa timp)
- LSTM → analizarea flow-ului video, imbinarea si analizarea frame-urilor in mod secvential.

In functie de rezultatele pe care le vom obtine in context-ul interfatarii, arhitectura finala este SUSCEPTIBILA la alte schimbari (adaugarea a alte metode de augmentare, modificat hiperparametrii).

De asemenea, numarul de clase folosit pentru labeling este susceptibil la schimbari.

Date + Procesare

Pentru task-ul pe care-l avem vom folosi o portiune din 20BNJester dataset, dataset care contine videoclipuri cu oameni executand diferite gesturi. Pentru un model stabil, care poate fi folosit pentru interfatare am folosit:

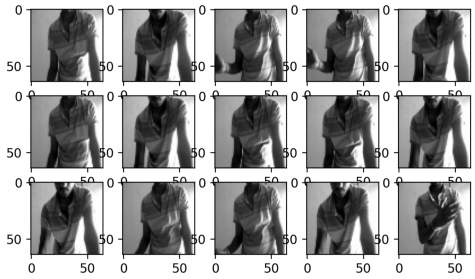
- Swipe Up

- Swipe Down
- Swipe Left
- Swipe Right
- No Gesture

Videoclipurile sunt luate fiecare, sunt unificate ca toate sa fie percepute ca seturi de 30 de frame-uri, dupa care imaginile sunt resized la rezolutie de 64×64, si convertite din RGB (reprezentare pe 3 canale) in grayscale (reprezentare pe un singur canal).

Ce poate fi imbunatatit? Pe langa procedeele deja existente, se mai poate adauga o forma de augmentare, si imaginile pot fi resize-uite la o rezolutie putin mai mare, dar in afara de asta, procesarea este in stadiu final.

20BNJester → https://www.kaggle.com/datasets/toxicmender/20bn-jester



Arhitectura CNN

Arhitectura aleasa este urmatoarea:

- Inputul (64×64×1)
- Conv3D1
  - Conv3D (32, (3, 3, 3)) + ReLU
  - MaxPooling3D (2, 2, 2) + ReLU
  - Dropout(0.25)
  - BatchNormalization()
- Conv3D2
  - Conv3D (64, (3, 3, 3)) + ReLU
  - MaxPooling3D (2, 2, 2) + ReLU
  - Dropout(0.25)
  - BatchNormalization()
- LSTM1
  - LSTM2D(40, dropout=0.1, return_sequence=true, padding=same)
- LSTM2

    - LSTM2D(20, dropout=0.1, padding=same)
- Dense layers
    - Dense(128)
    - Dense(4) + softmax

Epochs: 10

Optimizer: SGD (learning_rate = 0.01, momentum = 0.9)

De ce aceasta arhitectura?

De-a lungul a mai multor incercari, aceasta pare sa fie cea mai stabila arhitectura pentru metoda de procesare de date mentionata mai sus (fara augmentare). Se foloseste pentru ambele layere un kernel de 3×3×3, rezultatul caruia se duce intr-un MaxPooling de 2×2×2. Pentru acest tip de date, 2 layere par sa fie solutia buna, fiind balanta intre numarul de feature-uri invatat intr-un context 64×64 si numarul de clase posibile (adica 4). In ceea ce priveste numarul / structura layerelor de convolutie, este de asemenea susceptibila schimbarilor pentru cresterea rezolutiei sau folosirea altor metode de procesare.

Initial, exista un singur layer LSTM, dar dupa ce am adaugat inca unul s-a minimizat confuzia intre gesturi si recunoasterea pare mult mai fluida.

S-a folosit SGD in detrimentul Adam (care a fost folosit initial), pentru ca pare sa dea rezultate mai bune in context-ul generalizarii unor gesturi noi.

In cazul problemelor de overfitting, o sa fie nevoie de implementare a tehincilor de augmentare.

De asemenea, pentru a preveni overfitting, se folosesc Dropout(0.25) si BatchNormalization.

Benchmarks