

# Rapport: Emotion Detection med Deep Learning\*\*

## Introduktion

Detta projekt syftar till att utveckla en modell för att identifiera och klassificera ansiktsuttryck i realtid med hjälp av deep learning. Modellen kan känna igen sju olika känslor: Arg, Avsky, Rädska, Glad, Ledsen, Överraskad och Neutral. För detta ändamål användes ett Convolutional Neural Network (CNN) som tränades på ett dataset med bilder av ansikten märkta med olika känslor.

## Dataset

Datasetet består av totalt 35,887 bilder fördelade på träningsdata, valideringsdata och testdata:

Träningsdata\*\*: 22,968 bilder

Valideringsdata\*\*: 5,741 bilder

Testdata\*\*: 7,178 bilder

Varje bild i datasetet är kategoriserad i en av de sju känslorna, vilket gör det möjligt för modellen att lära sig identifiera skillnader mellan dem.

## Modellarkitektur

Modellen som används är en djup neural nätverksmodell med flera konvolutionella lager följt av fullt anslutna (dense) lager och dropout-lager för att motverka överanpassning. Arkitekturen består av följande lager:

- 1. Konvolutionella lager:** Extraherar viktiga funktioner från bilderna.
- 2. Pooling-lager:** Minskar dimensionerna av de konvolutionella lagrens utdata.
- 3. Dense-lager:** Kombinerar funktionerna för att göra slutgiltiga förutsägelser.
- 4. Dropout-lager:** Används för att förhindra överanpassning genom att slumpmässigt stänga av neuroner under träning.

## python

```
nn_model = Sequential()
nn_model.add(Dense(100, activation='relu', input_dim=n_cols))
nn_model.add(Dropout(rate=0.2))
nn_model.add(Dense(50, activation='relu'))
nn_model.add(Dense(1, activation='sigmoid'))

nn_model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

## Träning och Utvärdering

Modellen tränades i 10 epoker. Under träningen övervakades både tränings- och valideringsnoggrannhet och förlust för att se hur väl modellen presterade. Efter varje epok registrerades noggrannhet och förlust för att bedöma modellens förbättring.

## **Resultat**

Modellen uppnådde följande prestanda på testdata:

- Träningsnoggrannhet: 56.16%
- Valideringsnoggrannhet: 52.93%
- Testnoggrannhet: 52.34%

Validerings- och testresultaten ligger nära varandra, vilket indikerar att modellen har generaliserat väl utan att överanpassa sig till träningsdata.

## **Realtids emotion detection**

En realtids emotion detection-applikation utvecklades med hjälp av OpenCV och den förtränaade modellen. Genom att använda webbkameran detekterades ansikten och klassificerades i realtid. Användaren kan även spara bilder med detekterade känslor genom att trycka på mellanslagstangenten.

```
python
if key == ord(' '):
    image_name = f"captured_emotion_{predicted_emotion}.jpg"
    cv2.imwrite(image_name, frame)
    print(f"Image saved: {image_name}")
```

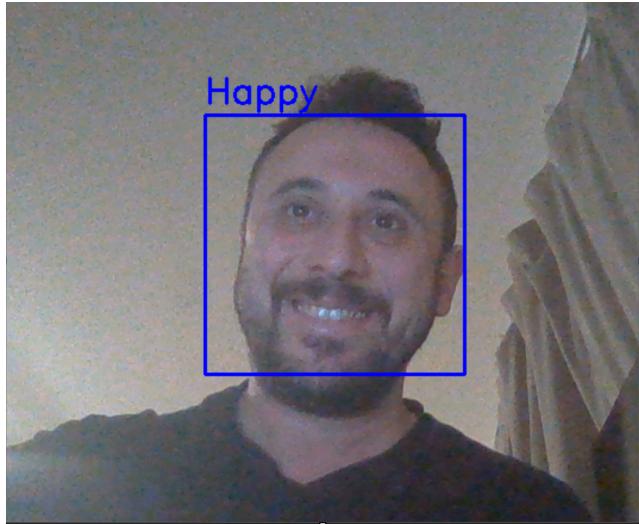
## **Slutsats och framtida arbete**

Projektet visade att det är möjligt att detektera känslor i realtid med hjälp av en CNN-modell. Modellen kan förbättras ytterligare genom att:

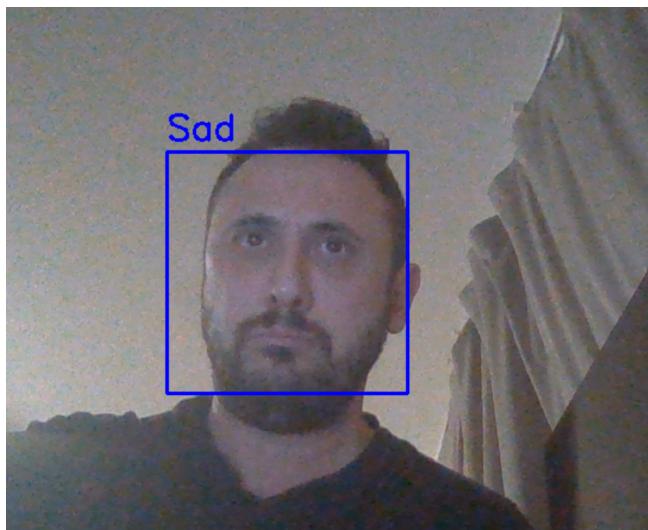
- Använda fler träningsdata med mer variation.
- Implementera mer avancerade arkitekturen som ResNet eller VGG16.
- Utforska andra regularisering tekniker för att ytterligare minska överanpassning.

## **Exempel på förutsägelser**

Här visas exempel på bilder tagna med systemet samt de förutsagda känslorna:



Happy



Sad