

Automated Labelling by Facial Recognition
Systemdokumentasjon
Versjon 1.0

REVISJONSHISTORIE

Dato	Versjon	Beskrivelse	Forfatter
21/05/23	1.0	1. Iterasjon	Stian, Matias og Markus

INNHOLDSFORTEGNELSE

1 INNLEDNING	4
2 ARKITEKTUR	5
3 PROSJEKTSTRUKTUR	6
4 KLASSEDIAGRAM	7
5 INSTALLASJON OG KJØRING	8
6 DOKUMENTASJON AV KILDEKODE	9

1 INNLEDNING

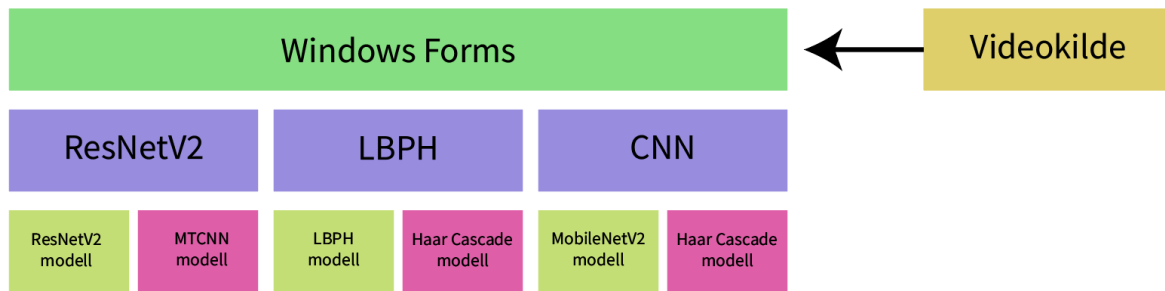
Dette dokumentet utgjør systemdokumentasjonen for programvaren utviklet som en del av gruppens bachelorprosjekt. Det er skrevet for å gi en omfattende og strukturert oversikt over systemet og implementeringen av programmet.

Hensikten med dokumentet er å dokumentere systemet på en klar og forståelig måte, slik at det blir presentert hvordan programmet fungerer. Dette inkluderer tekniske aspekter som arkitektur, komponenter og grensesnitt.

Dette dokumentet innebærer:

- Arkitektur
En oversikt over systemets struktur, komponenter, moduler og integrasjoner
- Prosjektstruktur
Mappe- og filstruktur for programvaren, samt kildekode og programvarebibliotek
- Klassediagram
Et diagram over klassestruktur
- Installasjon og kjøring
En beskrivelse av installasjon av programvaren og hvordan man kjører det
- Dokumentasjon av kildekoden
En beskrivelse av kildekoden.

2 ARKITEKTUR

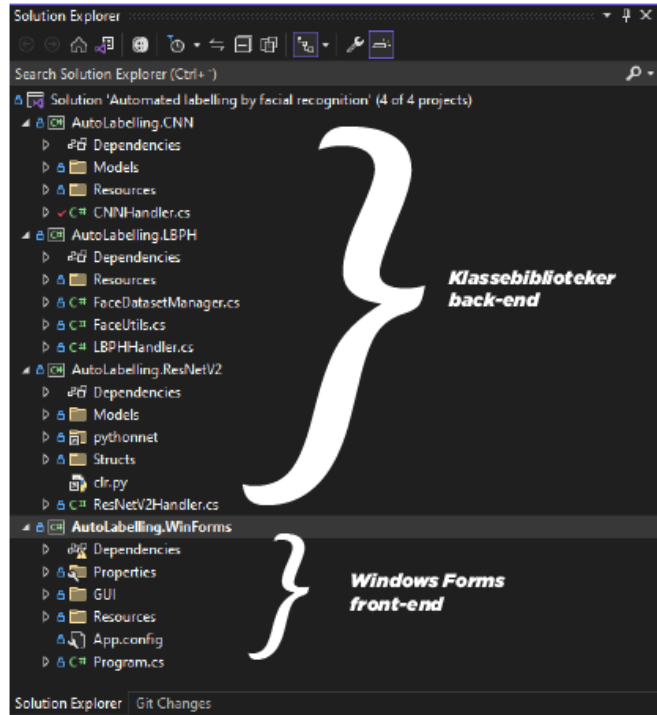


Figur 2.1: Skisse av applikasjonens arkitektur.

Windows Forms delprosjektet fungerer som et presentasjonslag for de tre underliggende klassebibliotekene, som hver tar i bruk modeller for ansiktsdetektering og ansiktsgjenkjenning. Valg av videokilde gjøres i Windows Forms delprosjektet, og hvert klassebibliotek bruker denne.

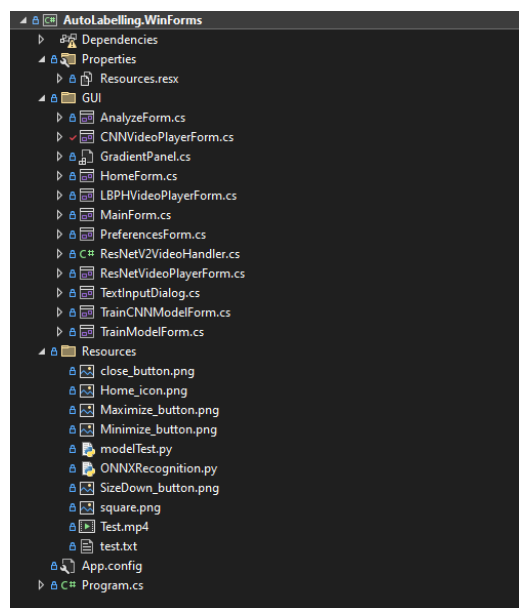
3 PROSJEKTSTRUKTUR

Figur 3.1 viser prosjektstrukturen til programvaren delt opp i moduler. De tre øverste prosjektetene refererer til back-end-komponenten i form av klassebibliotek, og det nederste prosjektet refererer til front-end-komponenten som tar i bruk de tre modellene over.



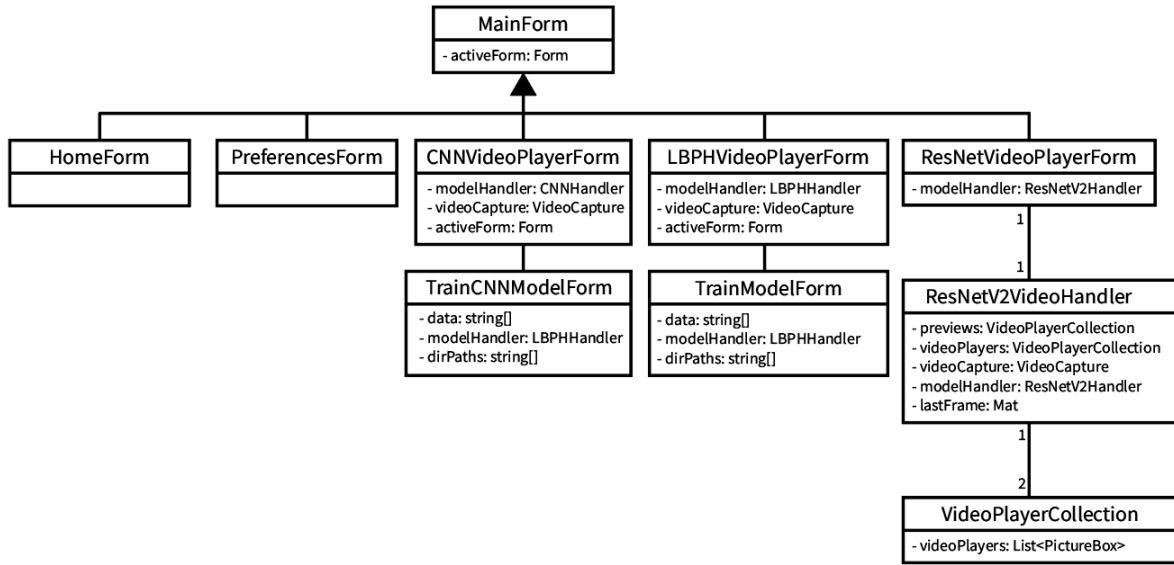
Figur 3.1: Viser oppdelingen av back-end komponenten i form av klassebibliotek og frontend komponenten i form av Windows Forms

I AutoLabelling.WinForms prosjektet er det mapper oppdelt i GUI og Resources, hvor GUI inneholder alle Forms for programvaren som vist under.

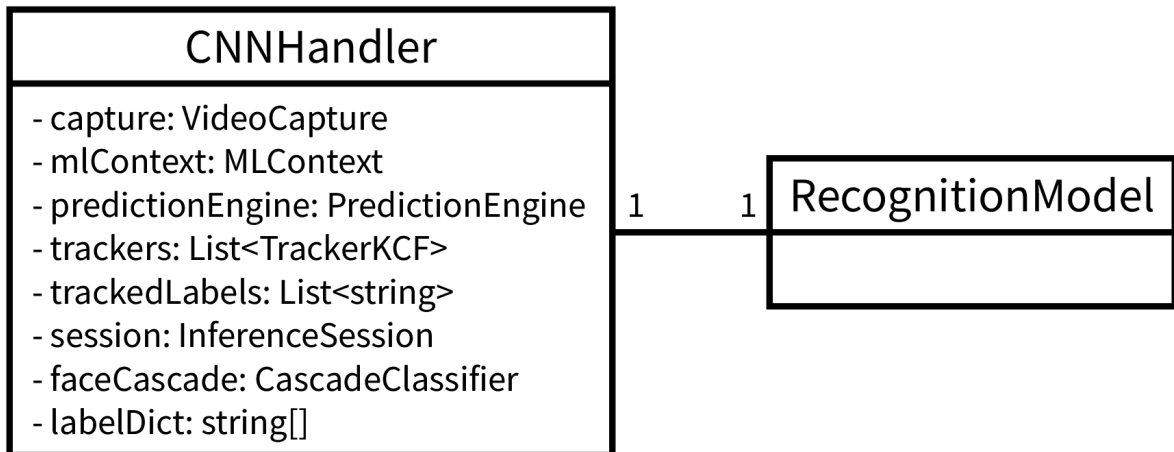


Figur 3.2: Oppdeling av forms og ressurser i Windows Forms prosjektet

4 KLASSEDIAGRAM



Figur 4.1: Enkelt klassediagram for Windows Forms delprosjektet.



Figur 4.2: Enkelt klassediagram for CNN klassebiblioteket.

5 INSTALLASJON OG KJØRING

For installasjon av programmet, pakk ut vedlagt zip fil. Prosjektet i zip-filen, skal åpnes i Visual Studio (på en windows maskin), og kompiles derfra. Dette vil gi en executable som kan kjøres. Nødvendig packages for C# koden er vedlagt i prosjektet, så ingen videre installasjon er nødvendig.

For å kunne kjøre prosjektet, og trene en modell, må maskinen ha Python 3.8 installert fra hjemmesiden til Python. Etter at dette er installert, må flere Python dependencies installeres dersom disse ikke er installert fra før. Dette kan gjøres fra terminal ved å skrive følgende kommandoer:

```
pip install numpy==1.21.2
pip install tensorflow==2.6.0
pip install keras==2.6.0
pip install opencv-python==4.5.3.56
pip install scikit-learn==0.24.2
pip install tf2onnx==1.8.5
```

Visse typer eksterne kameraer kan ha problemer med kompatibilitet. For å løse dette anbefales det å installere nødvendig drivere for kameraet.

6 DOKUMENTASJON AV KILDEKODE

Dokumentasjon av kildekode har blitt brukt som et viktig verktøy for å enklere kunne forstå koden. Det har blitt lagt vekt på at dokumentasjon skal være forklarende og leselig, slik at det vil være enklere for eventuelle andre utviklere å utføre videre arbeid. For dokumentering av metoder, har XML dokumentasjon blitt brukt. All dokumentasjon har blitt skrevet på Engelsk, da dette er en standard for programutvikling.

```
/// <summary>
/// Adds a person to the table, and readies the person to be trained on.
/// Uses the pictures added and the name added to create a raw data folder for the person.
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void AddPerson_button_Click(object sender, EventArgs e)
{
```

Figur 6.1: Eksempel på XML dokumentasjon i prosjektet

Kommentarer har også blitt brukt for å beskrive visse deler av koden. Denne typen dokumentasjon har blitt brukt for å forklare hensikten til en variabel eller for å tydeliggjøre spesielt tekniske deler av koden.

```
def create_model(num_classes):
    # Load the MobileNetV2 model pre-trained on the ImageNet dataset
    base_model = MobileNetV2(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

    # Freeze the layers of the base model so they are not trainable
    for layer in base_model.layers:
        layer.trainable = False

    # Add a global average pooling layer to reduce the spatial dimensions of the output
    x = base_model.output
    x = GlobalAveragePooling2D()(x)

    # Add two dense layers with ReLU activation functions
    x = Dense(1024, activation='relu')(x)
    x = Dense(512, activation='relu')(x)
```

Figur 6.2: Forklarende kommentarer for deler av kildekode

