

# Handleiding

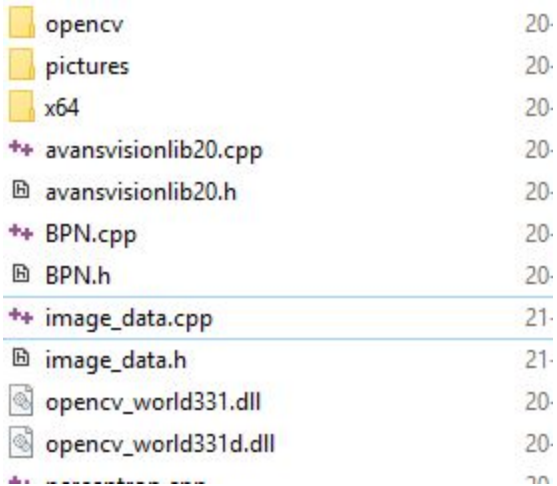
# **Neuraal Netwerk Applicatie**

---

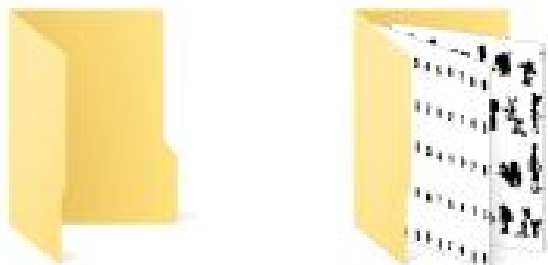
Jeroen van Os en Remco Sannen  
21-12-2017

Voor het eerste gebruik

Voor het opstarten van de applicatie, is het belangrijk om te controleren of OpenCV 3.3.1 in de applicatie geïnstalleerd is. De reden dat het in de applicatie zelf geïnstalleerd is, is voor de overdraagbaarheid hiervan. Ook is het belangrijk een map in C:/ te maken genaamd neuralnetwork, met hierin twee mappen: test en trainingsset. In trainingsset worden de afbeeldingen waarop getraind wordt geplaatst, in test de afbeeldingen waarmee je wilt kijken wat voor object het is. Bij het gebruik van de camera is het noodzakelijk de camera (de uEye van IDS) aan te sluiten.



opencv	20
pictures	20
x64	20
++ avansvisionlib20.cpp	20
avansvisionlib20.h	20
++ BPN.cpp	20
BPN.h	20
++ image_data.cpp	21
image_data.h	21
opencv_world331.dll	20
opencv_world331d.dll	20
-----	20



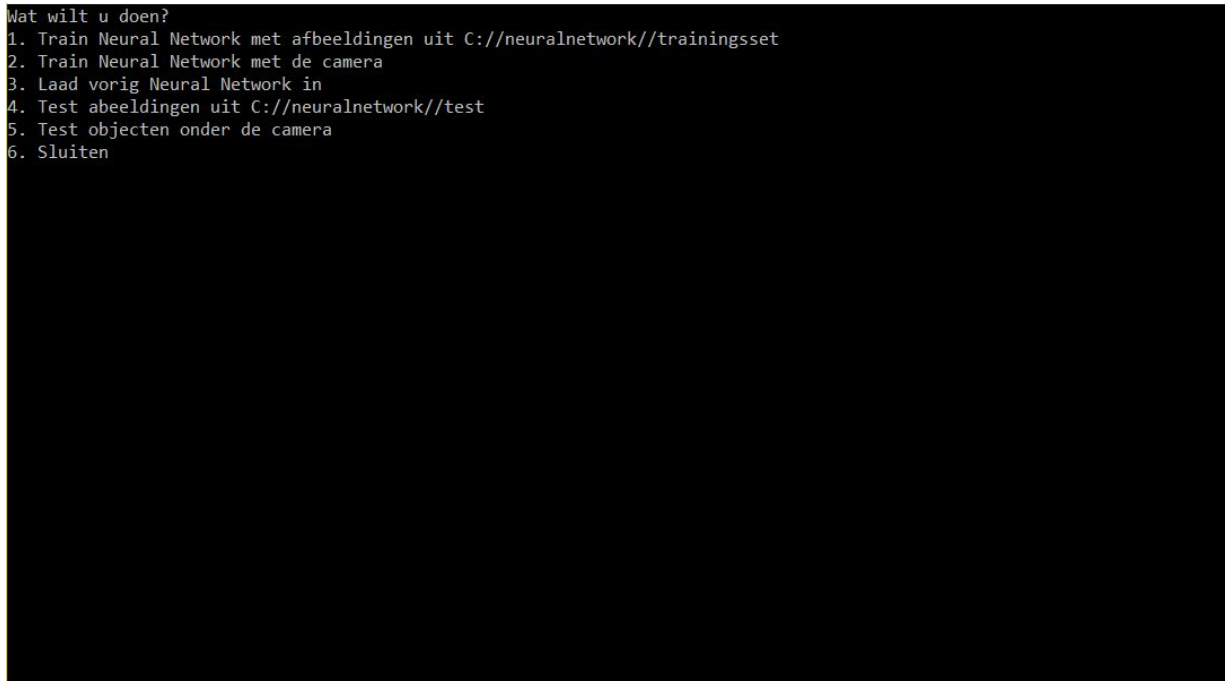
test                      trainingsset

**Afbeelding 1:** Opencv 3.3.1 aanwezig in het project

**Afbeelding 2:** Mappenstructuur in C:/neuralnetwork/

## Het gebruik van de applicatie

Bij het opstarten van de applicatie krijgt u een keuzemenu te zien (Afbeelding 3).

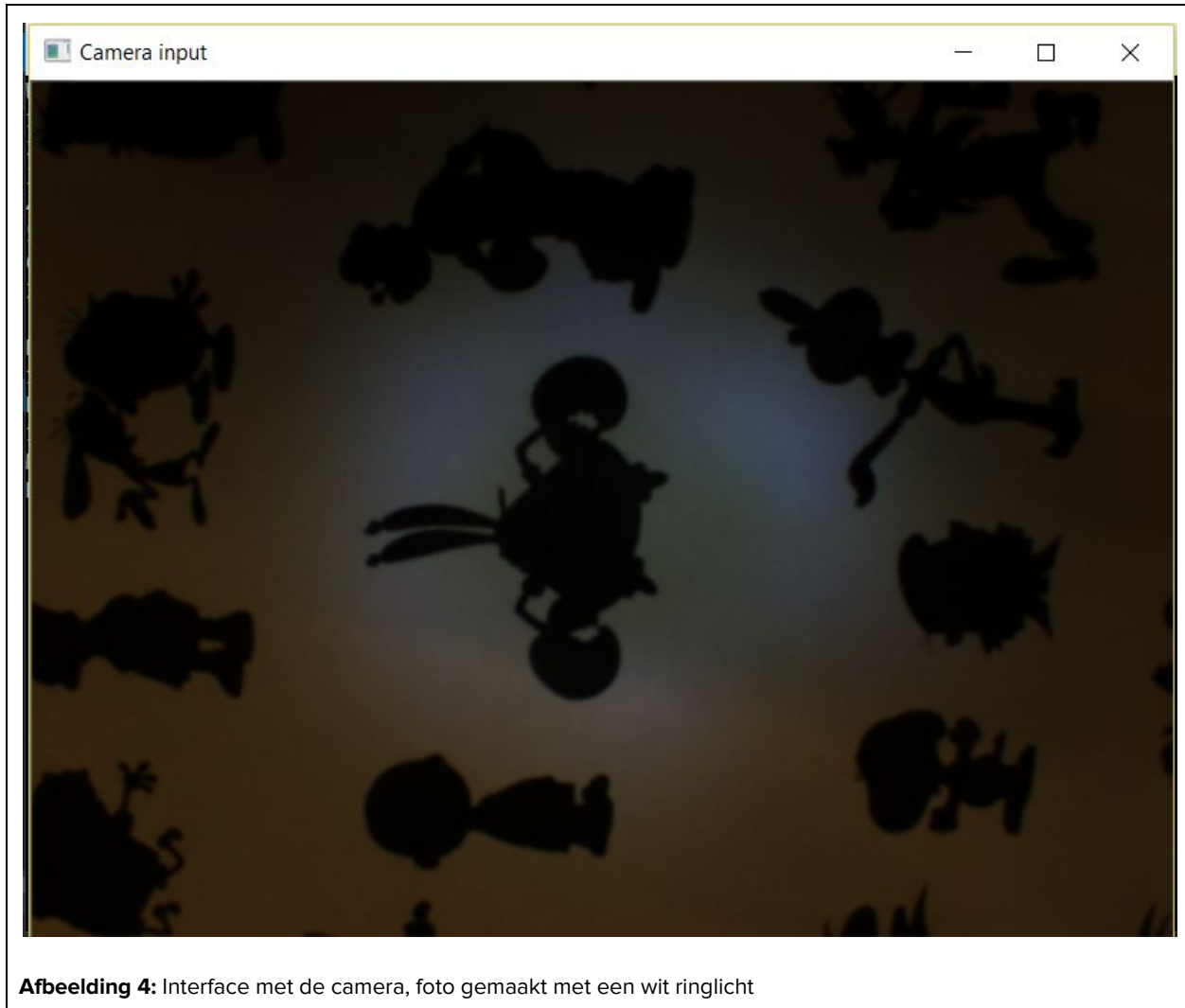


```
Wat wilt u doen?  
1. Train Neural Network met afbeeldingen uit C://neuralnetwork//trainingsset  
2. Train Neural Network met de camera  
3. Laad vorig Neural Network in  
4. Test afbeeldingen uit C://neuralnetwork//test  
5. Test objecten onder de camera  
6. Sluiten
```

**Afbeelding 3:** Keuzemenu

In dit menu zijn meerdere mogelijkheden te kiezen. Zo is het mogelijk het Neuraal Netwerk te trainen, de gegevens van een vorige keer in te laden en afbeeldingen testen op geleerde onderdelen. Het trainen en het testen kan met bestaande afbeeldingen en met de camera.

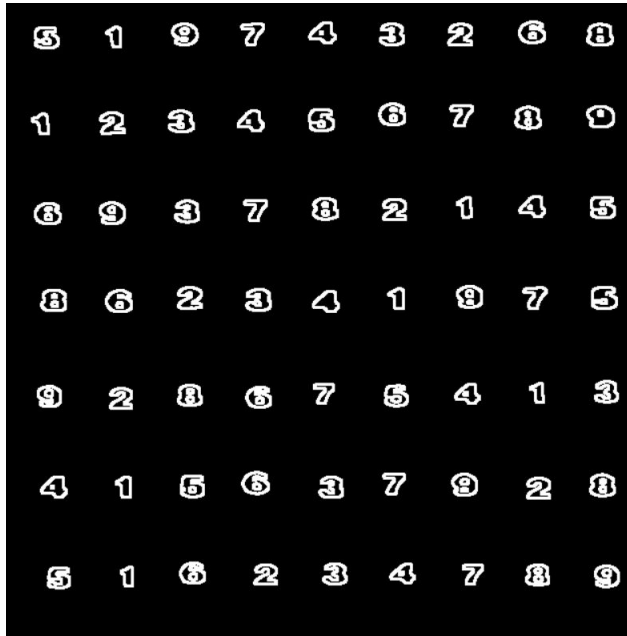
Als u voor de optie met de camera kiest, krijgt u het beeld hiervan te zien (Afbeelding 4). Met dat scherm actief (ook wel geselecteerd), kunt u met spatiebalk foto's maken. U kunt er zoveel maken als nodig lijkt. Als u daarna op escape (esc) drukt, gaat u met de genomen foto's naar de volgende stap



**Afbeelding 4:** Interface met de camera, foto gemaakt met een wit ringlicht

Nadat deze foto's ingeladen of gemaakt zijn, komt u naar het trainen of het testen. Als eerste zal het trainen behandeld worden.

Bij het trainen worden 2 afbeeldingen geopend. Een van de afbeeldingen zijn alle gevonden contouren (Afbeelding 5), de andere is het huidige contour wat geassocieerd moet worden (Afbeelding 6). In de console (terminal/tekstvenster) kunt u invoeren wat voor object het is (Afbeelding 7). Bij het invoeren van 'x' zal hij het contour negeren. Zo kunt u het Neuraal Netwerk specifiek trainen. Het is mogelijk om een of meerdere contouren per afbeelding/foto te classificeren.



**Afbeelding 5:** Alle contouren uit de afbeelding



**Afbeelding 6:** Het huidige contour

```
Wat wilt u doen?
1. Train Neural Network met afbeeldingen uit C://neuralnetwork//trainingsset
2. Train Neural Network met de camera
3. Laad vorig Neural Network in
4. Test afbeeldingen uit C://neuralnetwork//test
5. Test objecten onder de camera
6. Sluiten
1

Hoeveel iteraties? Standaard is 10000
10000
Hoeveel hidden neuronen in de layers? Standaard is 200
200
Loading images from folder C://neuralnetwork//trainingsset
Welk type object is dit? X voor overslaan
cijfer
```

**Afbeelding 7:** Classificatie in de terminal

Na het trainen van het Neuraal Network, zal het laten zien wat zijn getrainde klassen zijn, en hoe nauwkeurig het was bij het trainen (Afbeelding 8). Het Neuraal Network zal automatisch opgeslagen worden, zodat het later opnieuw ingeladen kan worden.

```
Uitkomst na training: [2] Verwachte Output: [0] (5)
Uitkomst na training: [4] Verwachte Output: [2] (9)
Uitkomst na training: [5] Verwachte Output: [3] (8)
Uitkomst na training: [2] Verwachte Output: [4] (7)
Uitkomst na training: [4] Verwachte Output: [8] (6)
Uitkomst na training: [1] Verwachte Output: [0] (5)
Uitkomst na training: [0] Verwachte Output: [1] (1)
Uitkomst na training: [5] Verwachte Output: [3] (8)
Uitkomst na training: [6] Verwachte Output: [5] (2)
Uitkomst na training: [6] Verwachte Output: [7] (3)
Uitkomst na training: [1] Verwachte Output: [4] (7)
Uitkomst na training: [5] Verwachte Output: [2] (9)
Uitkomst na training: [5] Verwachte Output: [6] (4)
Uitkomst na training: [5] Verwachte Output: [8] (6)
Accuracy:
0.82528

Saving Neural Network at: C://neuralnetwork//mlp.yaml
```

**Afbeelding 8:** Uitkomst training Neuraal Network

Voor het testen van afbeeldingen of foto's, zal het wederom het contour laten zien, en weergeven wat het Neuraal Network denkt wat voor klasse het is. Het zal eerst het index van de classificatie laten zien, gevolgd door de naam van de klasse (Afbeelding 9).

```
Uitkomst: [1] (1)
Uitkomst: [4] (7)
Uitkomst: [2] (9)
Uitkomst: [5] (2)
Uitkomst: [5] (2)
Uitkomst: [6] (4)
Uitkomst: [2] (9)
Uitkomst: [4] (7)
Uitkomst: [5] (2)
Uitkomst: [2] (9)
Uitkomst: [4] (7)
Uitkomst: [1] (1)
Uitkomst: [0] (5)
Uitkomst: [5] (2)
Uitkomst: [6] (4)
Uitkomst: [6] (4)
Uitkomst: [1] (1)
Uitkomst: [5] (2)
Uitkomst: [5] (2)
Uitkomst: [5] (2)
```

**Afbeelding 9:** Uitkomst classificatie test van contour