

## Inhoud

<b>Kleurrijke Morsecode: Innovatieve Generatie en Decodering .....</b>	<b>2</b>
Het ontstaan van het project .....	2
<b>De puzzelvraag is: <i>wat staat hier</i>.</b> .....	<b>2</b>
Wat is ChatGPT.....	2
Eerste experimenten.....	3
ChatGPT lost het niet op .....	4
Opdelen in kleine stappen .....	4
Van decoder naar encoder.....	6
Klopt het eerder gevonden decoder resultaat wel? .....	6
Encoder test .....	7
Uitbreiding .....	8
De encoder.....	9
OTC_enc_morse_kleuren_afbeelding.png.....	9
De decoder.....	9
OTC_dec_1_gedetectedeerde_cirkels.png .....	10
OTC_dec_2-cirkels_kleur_markering.png .....	10
OTC_dec_3_reconstructie_kleuren_cirkels.png .....	10
OTC_dec_4_ontcijferd_bericht.txt.....	10
Hoe kun je de Python bestanden gebruiken.....	11
Programma voor Windows, Mac en Linux.....	13
<b>Stap 1: Installeren van PyInstaller</b> .....	<b>13</b>
<b>Stap 2: Uw Python-script omzetten naar een uitvoerbaar bestand</b> .....	<b>13</b>
<b>Optionele stappen:</b> .....	<b>14</b>
<b>Opmerkingen:</b> .....	<b>14</b>
programma gebruik.....	15
Demo met verkeerde afbeeldingen .....	17

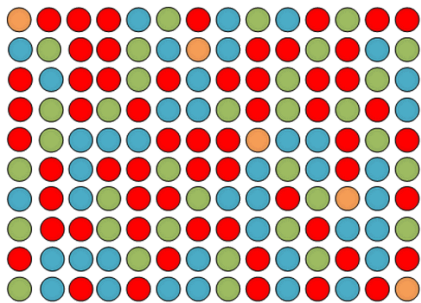
## Kleurrijke Morsecode: Innovatieve Generatie en Decodering

Will je gelijk naar gebruiksklare programma's, ga naar 'programma gebruik' onderaan het document.

### Het ontstaan van het project

Het idee voor dit project is ontstaan door de Kerst Puzzel 2023 van het OTC. De kerst periode zorgt voor een aantal vrije dagen. Dan is er wat extra vrije tijd. Tijd om de puzzel op te lossen. Dat is makkelijker gezegd dan gedaan. Dit is de eerste keer dat ik mee doe. Vorig jaar heb ik een bekende geholpen met wat vragen, maar nu is het tijd om het zelf te doen.

De puzzelvraag is: *wat staat hier.*



Deze afbeelding is de puzzel die opgelost moet worden.

Er is geen herkenbare structuur op de afbeelding te ontdekken. Misschien kan ik de hulp van een computer gebruiken om patronen te herkennen.

Omdat AI de laatste tijd veel aandacht krijgt, dacht ik om eens te onderzoeken of ChatGPT dit kan oplossen.

### Wat is ChatGPT

ChatGPT is een variant van de GPT (Generative Pre-trained Transformer) taalmodellen ontwikkeld door OpenAI. AI is '**kunstmatige intelligentie**'. Het is speciaal ontworpen om in staat te zijn natuurlijk ogende en samenhangende tekstuele gesprekken te voeren, wat het een ideaal model maakt voor toepassingen zoals chatbots en interactieve digitale assistenten.

Kenmerken van ChatGPT zijn onder meer:

1. **Taalbegrip en -generatie:** Het model is getraind op een grote dataset van tekst en kan zodoende complexe taalvragen begrijpen en beantwoorden in een natuurlijke, mensachtige manier.
2. **Veelzijdigheid:** ChatGPT kan worden gebruikt voor een breed scala aan toepassingen, van het beantwoorden van algemene kennisvragen tot het ondersteunen van technische troubleshooting, het schrijven van creatieve teksten, het assisteren bij coderingsproblemen, en meer.
3. **Voortdurend Leren:** Hoewel het model getraind is op een statische dataset, kan het worden geïntegreerd met systemen die het in staat stellen om te 'leren' van nieuwe interacties en feedback, waardoor het in de loop van de tijd verbetert.
4. **Interactie in Menselijke Stijl:** ChatGPT is ontworpen om conversaties te voeren die lijken op die tussen mensen, waardoor het een natuurlijke en gebruiksvriendelijke keuze is voor conversatie-interfaces.

ChatGPT maakt deel uit van een nieuwe generatie AI-modellen die de manier waarop we interactie hebben met machines en toegang hebben tot informatie kunnen veranderen, dankzij hun vermogen om menselijke taal te begrijpen en te genereren op een manier die voorheen niet mogelijk was.

## Eerste experimenten

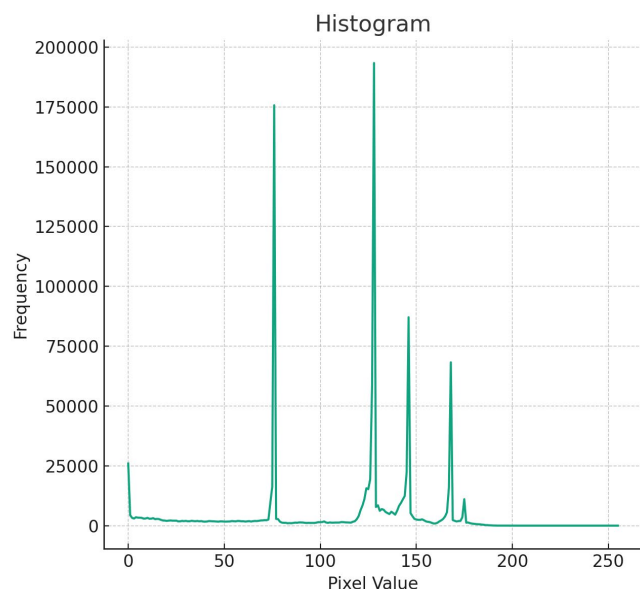
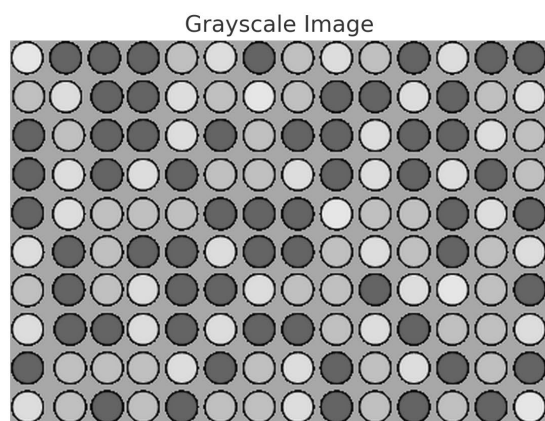
Het eenvoudigste is om de gegeven afbeelding in de gebruikte AI te stoppen. Dan komt het antwoord er zo uitrollen. Je upload het bestand en vraagt of er een bericht in te herkennen is. De vraag die je kunt stellen is of er een patroon in de afbeelding is te vinden. Met als extra input dat het een puzzel is en dat er een boodschap in is verstopt. Het resultaat was teleurstellend. ChatGPT kon niet vertellen wat het is of wat er staat. Ik kreeg alleen maar vragen i.p.v. antwoorden over wat er in zou kunnen staan.

Ok, dit kan dus niet met de huidige generatie AI. Dat komt vast wel in de toekomst. Uit nieuwsgierigheid wat ChatGPT (AI) allemaal kan ben ik verder gaan onderzoeken. Misschien kan ChatGPT me helpen in de oplossing vinden.

### ! Error analyzing ▾

De eerste vraag die ik stel resulteert in met de volgende analyse.

. Vervolgens komt het



Het is de afbeelding in grijs tinten gaan vertalen en dan de grijswaarden in een histogram gezet. Je ziet 4 hoofd tinten.

Het viel me op dat als ik een vraag aan ChatGPT stel, er vaak een draaiend cirkeltje is die aangeeft dat hij aan het analyseren is, of websites af scant. En vervolgens met een reactie komt. Er staat soms ook een klein blauw icoontje . Als ik daar de muis

View Analysis



boven beweegt is het volgende te zien.

Als ik daar op klik, zie ik dat er een heel script is gegenereerd. Een python code. Wat kan of moet ik daar mee?

There was an error generating a response

Regenerate

Helaas liep ik regelmatig tegen deze melding aan:

Dan is de Chat sessie vastgelopen en moet je een nieuwe starten. Met als gevolg dat de AI de informatie uit de andere Chat sessie kwijt is.

## ChatGPT lost het niet op

Zoals je ziet, ging het niet soepel. Na vele uren proberen en vele malen een soort van vastlopen van de AI engine, heb ik besloten het over een andere boeg te gooien. Mijn programmeer vaardigheden zijn op z'n zachts gezegd niet goed. Ik ben zo'n prutser die het werkend probeert te krijgen zonder een goede software ontwikkel basis. Een complete decoder bouwen in software, is niet voor mij weggelegd! Maar misschien kan ChatGPT me wel helpen met de juiste code genereren. Ik wil proberen het met Python op mijn pc op te lossen. Geholpen met hints van ChatGPT. Die combinatie blijkt bij mij veel beter te werken. Het is duidelijk dat ik het probleem of eigenlijk de oplossing, moet proberen op te hakken in hele kleine stappen.

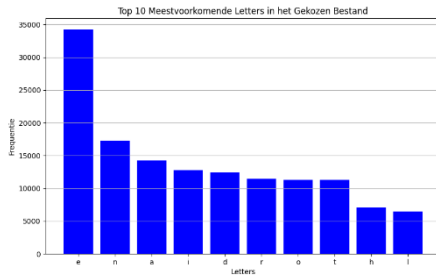
## Opdelen in kleine stappen

Ik zie een afbeelding met gekleurde bolletjes. Ieder bolletje heeft een dunne zwarte rand. Ik wil de bolletjes in de afbeelding proberen te vinden. Daarna de kleur proberen te detecteren. Dan de gegeven afbeelding van de OTC website proberen na te maken in Python. Ik zie dat het vier kleuren zijn. Maar wat kan er mee geduid worden? Ik weet het niet. Je moet ergens beginnen, maar waar.

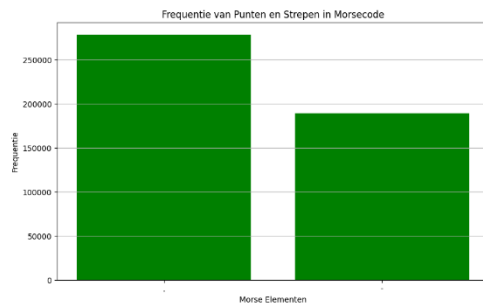
Het is een puzzel voor en door Radio amateurs. Stel dat het iets met morse te maken heeft. Hoe zou je het dan kunnen aanpakken. Je hebt vier verschillende kleuren. In morse heb je de 'punt', 'streep', 'spatie' en 'nieuw woord'. Dat zouden vier kleuren kunnen zijn. Als je het aantallen bolletjes per kleur in de afbeelding telt, kun je zien dat de meeste bolletjes rood zijn, daarna komt blauw het vaakst voor, dan groen en als minste de oranje.

De aantal punten en strepen zijn gelijk verdeeld over alle karakters in het alfabet en de cijfers. Wat echter niet gelijk is, is de karakters die in een tekst voor komen.

De top vijf van meest voorkomende karakters in een boek van Baantje:



1. E: .
2. N: -.
3. A: .-
4. I: ..
5. D: -..



De punt is de meest voorkomende teken in deze tekst omgezet in morse, daarna de streep, dan de spatie en dan nieuw woord. Het verschil tussen aantal punten en strepen is niet erg groot en kan bij een ander boek ook omgekeerd zijn.

Als je moet gokken zal de statistiek zeggen dat de meest waarschijnlijke betekenis is:

1. Rood = .
2. Blauw = -
3. Groen = spatie
4. Oranje = nieuw woord

Als ik dit probeer in python, moet ik de kleuren zien te herkennen, en dan iedere kleur invullen met code 1,2,3 of 4. Als ik die achter elkaar zet krijg ik een morse code. Vervolgens maak je een morsecode vertaal tabel. A t/m Z en 0 t/m 9. Dan kun je de morse omzetten naar tekst.

De stappen:

1. Detecteer cirkels in de ingevoerde afbeelding en teken witte randen bij herkenning
2. Bepaal de matrix afmeting aan de hand van de gedetecteerde cirkels
3. Bepaal binnen iedere gevonden cirkel welke kleur er in zit.
4. Verwerk elke gedetecteerde cirkel
5. Vind dubbele kleurposities voor debuggen
6. Morsecode samenstellen en vertalen
7. Plot de gevonden morse code en tekst

De oplossing is dan snel gevonden.  
 VANUIT DALLINEWEER GELUKKIG NIEUWJAAR  
 De oplossing ingestuurd naar OTC.

## Van decoder naar encoder

De decoder is gemaakt om de kerstpuzzel afbeelding te kunnen ontcijferen. Verder kun je niets met de decoder, want dit is de enige morse kleuren afbeelding in deze vorm die ik ben tegengekomen. Het zou interessant zijn als er meerdere morse kleurenafbeeldingen zijn. Toen dacht ik dat het leuk is om ook een morse kleuren afbeelding generator te maken waar je zelf een eigen tekst in kunt voeren. Dan kan de decoder die weer ontcijferen. Daar ben ik gelijk enthousiast mee begonnen.

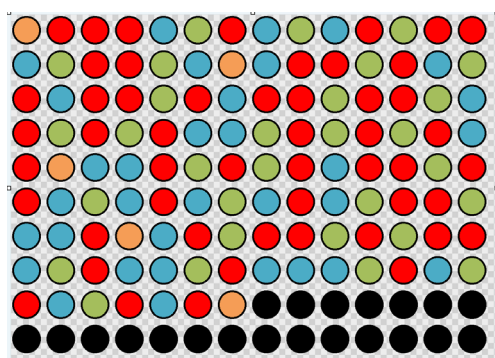
Gelukkig is de encoder wat makkelijker werkend te krijgen dan de decoder.

Het lukte in verhouding best snel om een 14 kolommen bij 10 rijen afbeelding te maken met kleuren bolletjes. Natuurlijk begin je bij het testen met slechts enkele karakters en langzaam ook wat woorden. Toen het leek te werken, heb ik de verkregen tekst uit de decoder “VANUIT ALLINEWEER GELUKKIG NIEUWJAAR “ in de encoder gestopt. De tekst word goed door de decoder gevonden. Dus een bevestiging van de goede werking. Alleen ziet de gegenereerde afbeelding er anders uit dan de OTC kerstpuzzel afbeelding. Wat heb ik fout in de generator/encoder?

Na lang zoeken en testen, kon ik nog steeds niets vinden in de encoder wat er fout kan zijn. Alle teksten die ik met de encoder maak worden goed gedecodeerd.

## Klopt het eerder gevonden decoder resultaat wel?

Toen begon ik te twijfelen.



Als ik de afbeelding van de OTC website vergeleek met de zelf gegenereerde afbeelding met de encoder, zie je dat het aantal kleuren bolletjes niet overeen komt.

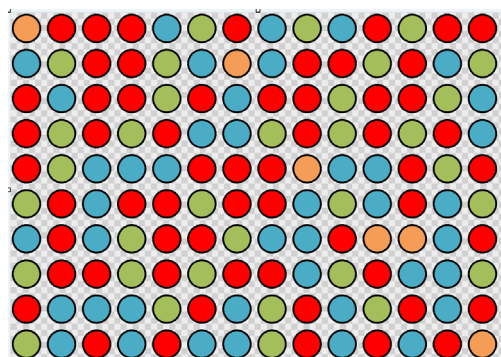
Dat duid erop dat er nog wat karakters missen! Ik realiseerde me ineens dat ik de speciale tekens niet in mijn code tabel heb gezet. Het zou toch niet zo zijn dat ik daarom toch nog wat heb gemist?

De proef op de som is om de decoder uit te breiden met de speciale leestekens. En dan kijken of dezelfde tekst er uit komt. De oorspronkelijke OTC afbeelding erin en.... Het resultaat was een net iets andere tekst dan de tekst die ik naar OTC had gestuurd. Dit was de tekst die eruit kwam.

VANUIT DALLINEWEER: GELUKKIG NIEUWJAAR!+

## Encoder test

Nu de speciale morse tekens ook toevoegen in de encoder. Dan zouden de afbeeldingen van de OTC kerstpuzzel en de encoder overeen moeten komen.



Hier de plot van de encoder.

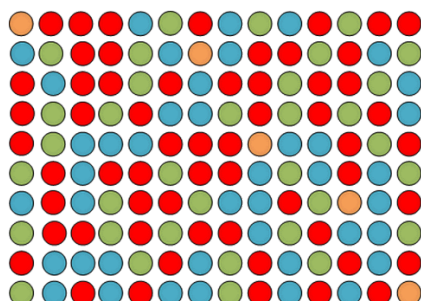
En de aantal bolletjes en kleuren in de zelf gegenereerde afbeelding is nu gelijk met de originele afbeelding. Dat geeft vertrouwen dat het nu helemaal klopt.

YES

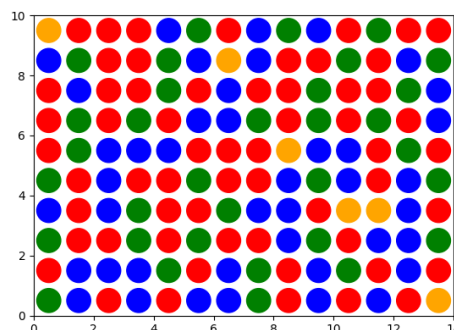
Oh jé, dat betekent dat mijn ingestuurde antwoord dus niet goed is. Daarop besloten de nieuwe tekst ook op te sturen naar OTC.

Het is duidelijk dat ik geen geoefend morse telegrafist ben. Als ik in de afbeelding zelf bolletje voor bolletje had gelezen en de bijbehorende tekst op papier had geschreven, was het me direct opgevallen dat er bijzondere morsecode karakters in staan. Maar dat heb ik nooit gedaan. Ik wilde het graag via de computer proberen.....

Zowel de encoder als de decoder hebben nu de uitgebreide morse code set (DICT). De gemaakte encoder kan nu getest worden met het eerder gevonden antwoord wat uit de decoder kwam als je er de OTC kerstpuzzel afbeelding in stopt. Namenlijk: 'VANUIT DALLINEWEER: GELUKKIG NIEUWJAAR!+'.



OORSPRONKELIJKE AFBEELDING



DECODER RECONSTRUCTIE AFBEELDING

Als ik de tekst in de encoder stop en dan weer in de decoder, komt er de gereconstrueerde afbeelding uit dat op één bolletje na exact gelijk is (afbeelding links van de OTC website, rechts van de decoder).

De kleuren van de door de decoder gereconstrueerde afbeelding zijn met opzet anders gemaakt. Het zijn de pure kleuren rood, blauw, groen en oranje. Dit is gedaan om te laten zien dat het echt zelf gegenereerd is en geen 'overlay'. De decoder slaat enkele tussenresultaten op in bestanden. Die afbeeldingen maken wel gebruik van 'overlays'.



## Uitbreiding

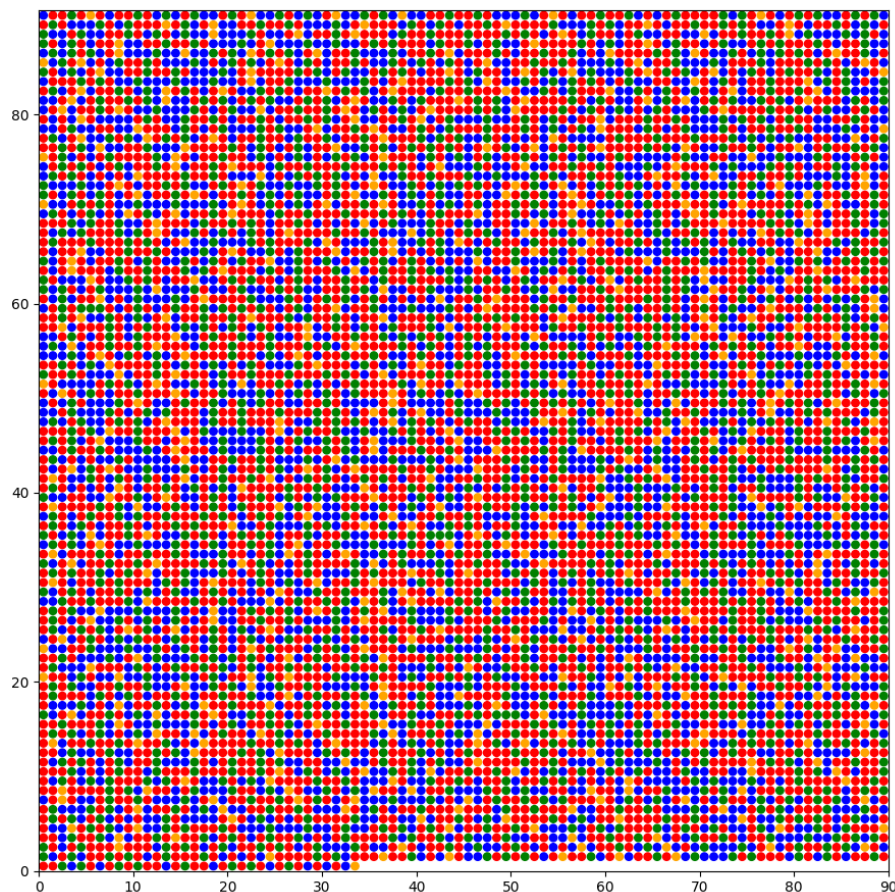
Met de encoder kun je tekst ingeven en die tekst wordt omgezet naar morse code. Daarna wordt ieder teken omgezet naar een kleur. Vervolgens in een 14 bij 10 matrix gezet en bolletje van gemaakt.

Bij de decoder gebeurt precies het omgekeerde.

Je kunt dus maximaal  $14 \times 10 = 140$  tekens (bestaande uit ., -, spatie, nieuw woord) gebruiken. Het zou ook leuk zijn om het meer dynamisch te maken. En de tekst grote onbeperkt te maken.

Daarom ben ik de encoder en decoder gaan aanpassen. De encoder probeert een vierkante matrix te maken. Dus evenveel rijen als kolommen. En de grote van die vierkant, hangt af van de hoeveelheid tekst. De decoder moet proberen de matrix grote te vinden en dan te decoderen.

Dat is nog een hele klus gebleken. Het is veel testen, uitproberen, doorzetten en overwinnen. Uiteindelijk lijkt het gelukt om de dynamische decoder zo te maken dat die tot wel 10 duizenden kleuren bolletjes in een afbeelding kan detecteren. Ieder bolletje vraagt enige processing om de bolletjes te vinden, de matrix afmeting te bepalen, de kleuren herkennen en te decoderen. Uiteraard is het zo dat de afbeeldingen met weinig gekleurde bolletjes snel gedecodeerd worden, en de afbeeldingen met zeer veel bolletjes veel trager zijn om te decoderen. Bij mij duurde het wel 1 minuut om een afbeelding met ruim 8 duizend bolletjes te decoderen.





Het genereren van deze afbeelding is bv 1 sec. en het correct lezen/decoderen/ontcijferen bijna 1 min.

Er zijn 2 uitvoeringen van de encoder en 2 uitvoeringen van de decoder. De OTC2023 bestanden zijn de eerste scripts die zijn gemaakt. De matrix afmeting is statisch met 14 bij 10. De OTC\_D bestanden zijn verder ontwikkeld en uitgebreider. Zie tabel hieronder.

Matrix afmeting	Type	Naam van het bestand	opmerking
14 bij 10	encoder	OTCP2023_encoder	snel, max 140 bollen
14 bij 10	decoder	OTCP2023_decoder	Traag, max 140 bollen
Dynamisch	encoder	OTC_D_encoder	snel, duizend+ bollen
Dynamisch	decoder	OTC_D_decoder	Zeer traag bij duizend+ bollen

## De encoder

De **encoder** genereert een bestand genaamd:

- **OTC\_enc\_morse\_kleuren\_afbeelding.png.**

### OTC\_enc\_morse\_kleuren\_afbeelding.png

Het bestand **OTC\_enc\_morse\_kleuren\_afbeelding.png** bevat een image/afbeelding met alle bolletjes. Deze kun je aan iemand geven. Net zoals in de OTC kerstpuzzel van 2023. Je kunt deze weer inlezen met de OTC\_D\_decoder en decoderen.

Voor het decoderen kun je altijd OTC\_D\_decoder gebruiken. De OTCP2023\_decoder zit er alleen bij omdat dit het script is waar ik voor het eerst een heb juiste antwoordt heb kunnen decoderen.

## De decoder

De **decoder** genereert 4 bestanden.:

- **OTC\_dec\_1\_gedetectede\_cirkels.png**
- **OTC\_dec\_2-cirkels\_kleur\_markering.png**
- **OTC\_dec\_3\_reconstructie\_kleuren\_cirkels.png**
- **OTC\_dec\_4\_ontcijferd\_bericht.txt**

Bestanden 1 t/m 3 zijn bedoeld als debug informatie tijdens de ontwikkeling, en kunnen helpen in het begrijpen van de werking van de decoder.

OTC\_dec\_1\_gedetectede\_cirkels.png

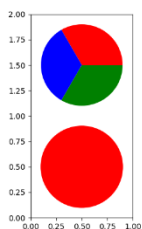
De eerste stap in het proces is het detecteren van de zwarte rand wat ieder bolletje heeft. Het bestand **gedetecteerde\_cirkels.png** bevat de oorspronkelijke afbeelding die je hebt geselecteerd, en daarbovenop wordt op iedere plaats waar de zwarte rand is herkend een witte cirkel overheen gezet. Zo kun je zien of de Canny detectie en de Hough transform goed is gaan.

OTC\_dec\_2-cirkels\_kleur\_markering.png

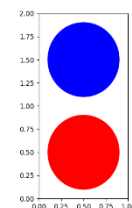
Het bestand **cirkels\_kleur\_markering.png** bevat ook de kleuren cirkels, maar dit keer met een kleine witte cirkel in iedere kleur die herkent is. Dan dan is te zien of

OTC\_dec\_3\_reconstructie\_kleuren\_cirkels.png

Als de decoder de tekst heeft gedetecteerd, wordt die op het scherm geprint en tegelijkertijd de geregenereerde bolletjes afbeelding getoond. Als het goed is zie je in ieder bolletje slechts één kleur. Als meerdere kleuren in één bol ziet, is er ergens iets fout gegaan.



Hier een voorbeeld (links) waar bv een letter n niet goed wordt gedetecteerd. De streep wordt door 3 kleuren herkend. En rechts is het resultaat na tunen van de juiste kleurinstellingen in het Python script. Ook andere fouten in de python code kunnen voor meerdere kleuren in één bolletje zorgen.



OTC\_dec\_4\_ontcijferd\_bericht.txt

Dit is een tekst bestand. Hier staat de morse code die is gevonden en de vertaling van de morse code naar tekst.

## Hoe kun je de Python bestanden gebruiken

Voor de Morsecode afbeelding generator en detector worden verschillende Python bibliotheken gebruikt. Hier is een overzicht van de benodigde bibliotheken en waarvoor ze gebruikt worden:

1. **NumPy (numpy)**: Een fundamentele bibliotheek voor wetenschappelijke berekeningen in Python. Het wordt gebruikt voor het maken en manipuleren van arrays, wat essentieel is voor het creëren en verwerken van de matrixrepresentaties van de Morsecode.

Installatie: `pip install numpy`

2. **OpenCV (cv2)**: Een krachtige bibliotheek voor beeldverwerking en computer vision. Het wordt gebruikt voor het genereren van afbeeldingen van de Morsecode en voor het verwerken van afbeeldingen in de detector.

Installatie: `pip install opencv-python`

3. **scikit-image (skimage)**: Een collectie algoritmen voor beeldverwerking. In deze code wordt het gebruikt voor functies zoals edge detection en Hough Circle Transform voor het detecteren van cirkels in de afbeeldingen.

Installatie: `pip install scikit-image`

4. **Pillow (PIL Fork, PIL)**: Een bibliotheek voor het openen, manipuleren en opslaan van verschillende bestandsformaten van afbeeldingen. Het wordt gebruikt voor basisbewerkingen met afbeeldingen, zoals het openen en converteren van afbeeldingen.

Installatie: `pip install Pillow`

5. **tkinter (tk)**: De standaard GUI (Graphical User Interface) bibliotheek voor Python. In deze code wordt het gebruikt voor het maken van een bestandsdialoogvenster om afbeeldingen te selecteren.

Tkinter is standaard inbegrepen in de meeste Python-installaties, dus je hoeft deze meestal niet apart te installeren.

6. **matplotlib (matplotlib)**: Een bibliotheek voor het creëren van statische, interactieve en geanimeerde visualisaties in Python. Het wordt gebruikt voor het plotten van kleurencirkels.

Installatie: `pip install matplotlib`

Als je alle benodigde bibliotheken wilt installeren, kun je de volgende commando's gebruiken:

```
pip install numpy opencv-python scikit-image Pillow matplotlib
```

Deze bibliotheken bieden een krachtige set van tools voor beeldverwerking en visualisatie, wat essentieel is voor het Morsecode afbeelding project.

Uitleg decoder:

### 1. Bestandsselectie en afbeeldingsverwerking

- o `select_file()`: Opent een dialoogvenster om een bestand te kiezen.
- o `cv2.imread()`: Leest het geselecteerde bestand.
- o `cv2.cvtColor()`: Zet de kleuren om naar HSV-formaat.
- o `detect_and_draw_circles()`: Detecteert cirkels in de afbeelding en slaat een afbeelding op met de gedetecteerde cirkels getekend als laag over de ingevoerde afbeelding.

### 2. Sorteren van Cirkel Coördinaten en Berekenen van Matrix Afmetingen

- o Sorteert de x- en y-coördinaten van de gedetecteerde cirkels.
- o Berekent het aantal unieke rijen en kolommen gebaseerd op drempelwaarden.

### 3. Initialisatie en Verwerking van Matrices

- o Initialiseert matrices voor kleurdetectie.
- o Verwerkt elke gedetecteerde cirkel en vult de matrices in.

### 4. Vinden van Dubbele Kleurposities

- o `find_double_colored_positions()`: Vindt posities in de matrices waar meerdere kleuren zijn gedetecteerd.

### 5. Markeren van Gedetecteerde Kleuren (voor debugging)

- o `mark_detected_colors()`: Markeert gedetecteerde kleuren in de afbeelding en slaat het resultaat op.

### 6. Morsecode Samenstellen en Vertalen

- o `compile_and_translate_morse_code()`: Stelt morsecode samen uit de matrices en vertaalt deze.

### 7. Output Secties

- o Printen naar command prompt: `print_command_prompt_output()`
- o Schrijven naar een bestand: `write_to_file()`
- o Tonen in een popup-venster: `show_in_popup_window()`

## Programma voor Windows, Mac en Linux

Je kunt een Python-script omzetten naar een uitvoerbaar bestand voor Windows (of andere besturingssystemen) met behulp van verschillende tools. Een van de meest populaire tools hiervoor is `PyInstaller`. Hier is een stapsgewijze handleiding om dit te doen met

`PyInstaller`:

Als je Python op je Windows, Mac of Linux computer installeert, zorgt Python er zelf voor dat het op jouw specifieke OS gaat werken. Dus op een Windows computer genereert het uitvoerbare bestanden voor je Windows computer, en op de Mac computer genereert Python automatisch uitvoerbare bestanden voor je Mac computer. Bij Linux idem. De werkwijze op alle drie computers is identiek.

### Stap 1: Installeren van PyInstaller

Eerst moet je `PyInstaller` installeren. Dit kan eenvoudig worden gedaan met `pip`:

Typ dit in de cmd window:

```
pip install pyinstaller
```

### Stap 2: Uw Python-script omzetten naar een uitvoerbaar bestand

Navigeer in de command prompt of terminal naar de directory waar uw Python-script zich bevindt. Voer vervolgens het volgende commando uit:

Typ dit in de cmd window:

```
pyinstaller --onefile your_script_name.py
```

- `--onefile`: Dit zorgt ervoor dat het uitvoerbare bestand als één enkel bestand wordt gemaakt.

Na het uitvoeren van dit commando zal `PyInstaller` een `dist` directory in de huidige map aanmaken. Binnen deze directory vindt je het uitvoerbare bestand met de naam `your_script_name.exe`.

## Optionele stappen:

- **Icon toevoegen:** Als u een aangepast pictogram aan uw uitvoerbare bestand wilt toevoegen, kunt je de `--icon` optie gebruiken:

Typ dit in de cmd window:

```
pyinstaller --onefile --icon=your_icon.ico your_script_name.py
```

- **Zonder consolevenster:** Als je niet wilt dat er een consolevenster wordt geopend wanneer jouw programma wordt uitgevoerd (bijvoorbeeld voor GUI-applicaties), kunt je de `--noconsole` optie gebruiken:

Typ dit in de cmd window:

```
pyinstaller --onefile --noconsole your_script_name.py
```

## Opmerkingen:

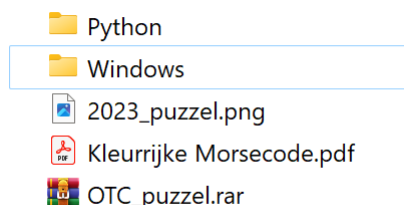
- Het uitvoerbare bestand zal groter zijn dan je misschien verwacht, omdat het alle benodigde Python-bibliotheken en een Python-interpreter bevat.
- Hoewel het `--onefile` argument handig is, kan het de opstarttijd van het uitvoerbare bestand verlengen. Als dit een probleem is, kunt je overwegen om deze optie weg te laten; dit zal echter resulteren in meerdere bestanden in plaats van één enkel uitvoerbaar bestand.
- Zorg er altijd voor dat je het uitvoerbare bestand test op de doelcomputer of -omgeving om te zorgen dat er geen afhankelijkheidsproblemen zijn.
- De `_internal` map met de benodigde Python-bibliotheken en een Python-interpreter kan gedeeld worden. Je mag van 2 of meerdere `_internal` mappen overelkaar heenschrijven, je mag maar hoeft niet te overschrijven als een bestand al bestaat. Je kunt dan alle gemaakte exe bestanden in één map zetten met de gedeelde `_internal` map.

Met deze stappen kunt je jouw Python-script omzetten naar een uitvoerbaar bestand voor je Windows of Mac computer.



## programma gebruik

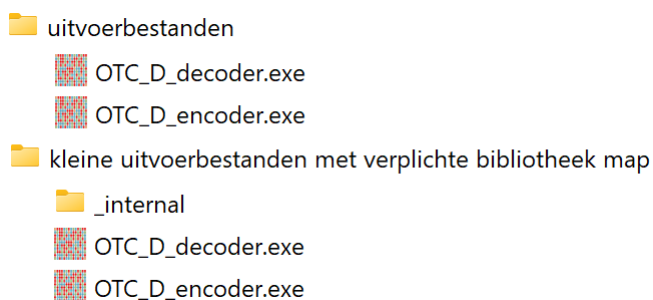
De programma's die reeds beschikbaar zijn, zijn voor Windows OS. Als je programma's voor Mac of Linux wilt hebben kun je ze zelf compileren op de hierboven beschreven methode.



Hier de hoofdmap:

In het rar bestand staat alles in deze en onderliggende mappen samengevoegd en gecomprimeerd. Dat is handig om te downloaden en op je eigen pc weer uit te pakken, i.p.v. bestand voor bestand downloaden.

In de Windows map zijn 2 sub-mappen te vinden. In de map '**uitvoerbestanden**' zijn twee bestanden die je kunt kopiëren en ergens op je pc kun zetten waar je dat wilt. Deze kun je opstarten zonder te hoeven installeren. Omdat deze alle bibliotheken al in het bestand hebben zitten, zijn ze iets groter en trager.



In de map '**kleine uitvoerbestanden met verplichte bibliotheek map**' staan weer twee uitvoerbare bestanden die de bijbehorende map **\_internal** altijd nodig heeft. Deze variant is wel wat sneller in uitvoering. Je mag zelf kiezen wat je voorkeur is.

De map **\_internal** bevat alle benodigde bibliotheken. Zonder die map zal het niet werken.

Zowel de encoder als decoder laten aan het eind van het uitvoerproces een afbeelding op scherm zien van de kleuren matrix. Zodra je deze weg klikt door op het kruisje rechts boven van de afbeelding te klikken, zal het programma afsluiten.

Op de plek waar de programma's staan, worden na het uitvoeren ervan onderstaande bestanden opgeslagen.

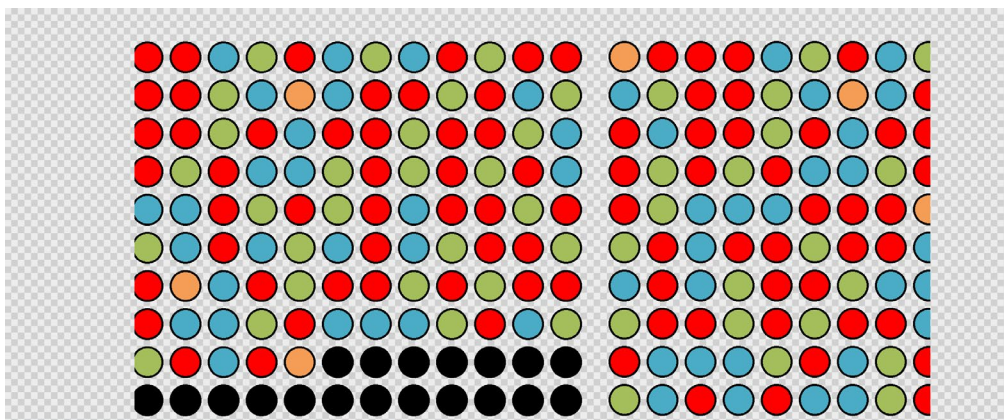
- OTC\_dec\_1\_gedetectedeerde\_cirkels.png
- OTC\_dec\_2-cirkeLOTC\_dec\_2-cirkels\_kleur\_markering.png
- OTC\_dec\_3\_reconstructie\_kleuren\_cirkels.png
- OTC\_dec\_4\_ontcijferd\_bericht.txt
- OTC\_enc\_morse\_kleuren\_afbeelding.png

De decoder maakt het bestand met **\_enc\_** in de naam, en de decoder maakt bestanden met **\_dec\_** in de naam.

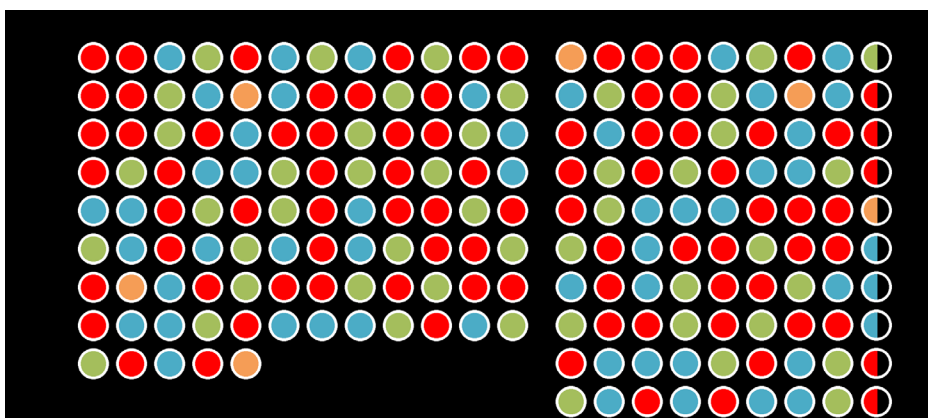
Uitleg over deze bestanden staat eerder in dit document. Klik op de bestandsnaam en je komt (als alles goed werkt) vanzelf bij de uitleg.

## Demo met verkeerde afbeeldingen

Hier een bewust verkeerd samengestelde afbeelding.

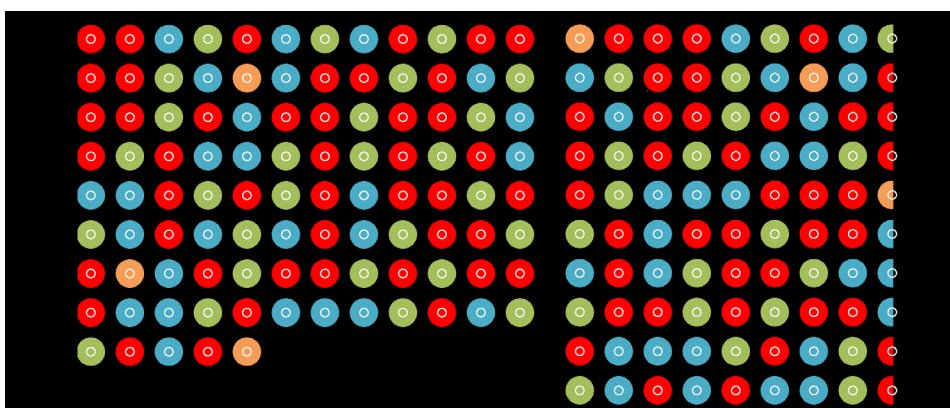


Deze wordt geladen in de decoder. Hieronder de output :**OTC\_dec\_1\_gedetecteerde\_cirkels.png**.



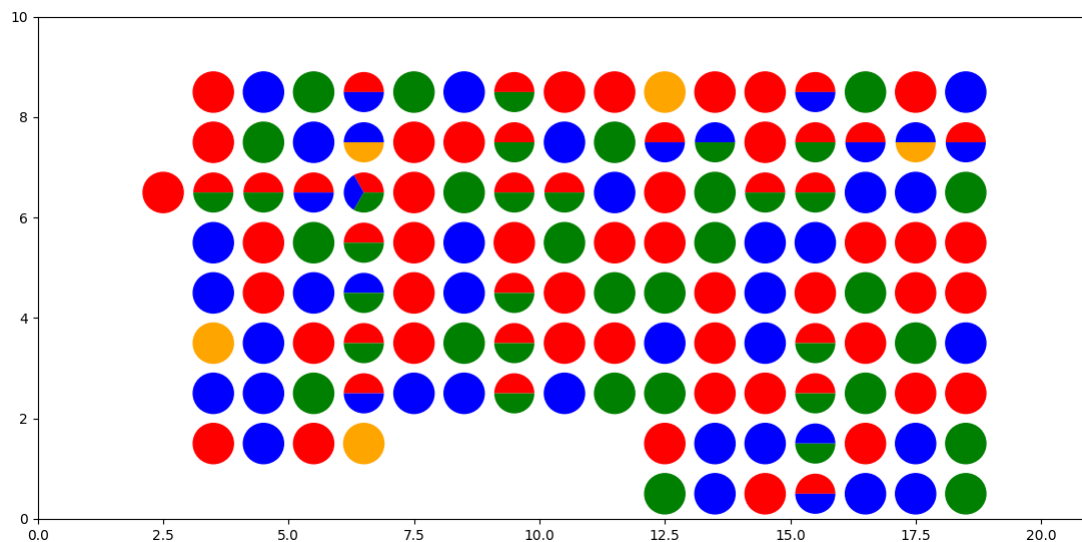
Deze afbeelding bestaat uit de bovenste afbeelding, met een extra laag eroverheen gezet met de gedetecteerde cirkels. Je ziet hier dat het is gelukt om alle cirkels te vinden. Zelfs de halve cirkels.

Hieronder de output van: **OTC\_dec\_2-cirkelOTC\_dec\_2-cirkels\_kleur\_markering.png**



Ook deze afbeelding bestaat uit de originele ingeladen afbeelding, met een extra over-layer van de gevonden kleuren binnen de cirkels. Alles lijkt tot dusver goed.

Hieronder de **OTC\_dec\_3\_reconstructie\_kleuren\_cirkels**.



Dit is een volledig door de decoder gegenereerde afbeelding. Dus geen 'overlay' meer. Extra zichtbaar gemaakt door de zuivere rood, blauw, groen en oranje kleuren te gebruiken.

Je ziet dat het hier niet meer goed gaat. De encoder verwacht een matrix, waar op alle cirkels op een gelijke afstand onder en naast elkaar staan. Het probeert een matrix te vinden en in te vullen. Er zijn cirkels te zien die door meerdere **verschillende kleuren tegelijkertijd** worden gezien. Dat is niet goed. Er mogen ook geen **gaten in de matrix** zitten.

Als je deze afbeelding ziet, weet je al dat het niet goed is. Er kan geen geldige tekst gegenereerd worden.

Dit voorbeeld kan helpen te begrijpen hoe de decoder is opgebouwd, en hoe de bestanden OTC\_dec\_1 t/m 3 kunnen helpen debuggen.