# Implementatieplan Edge Detection

## Namen en datum

Daniel Klomp 1661521  
Jop van Buuren 1658718

## Doel

Het doel van het project is om duidelijk te krijgen hoe de Egde Detection gaat worden geïmplementeerd. Daarbij worden de verschillende methode van Edge Detection vergeleken en wordt de beste optie uitgekozen om te worden verwerkt tot code. Hierbij is het belangrijk dat de Egde Detection duidelijk de randen (edges) van de afbeelding weet weer te geven.

## Methoden

Edge Detection kan op meerdere manieren worden geïmplementeerd. In dit hoofdstuk worden deze verschillende methoden uitgelegd, waardoor ze vervolgens met elkaar vergeleken kunnen worden.

Bij Methode 1 maken we gebruik van de Laplacian. Hierin wordt door middel van een 9x9 Mask de waarde van de pixels aangepast. Doordat bij het gebruik van de Laplacian filtering de waarden van de pixels onder de nul terecht kunnen komen, moeten de waarden na afloop worden verhoogd met +127 om alle edges goed zichtbaar te maken. Hierbij is het belangrijk dat de gebruikte afbeelding geen ruis bevat, omdat deze wordt versterkt door de Laplacian bewerking.

Bij Methode 2 maken we daar in tegenstelling ook nog gebruik van een Gaussian ruisvermindering in combinatie met de Laplacian. Hierdoor is het probleem van ruis na de Laplacian bewerking een stuk minder aanwezig. Hierbij is het belangrijk een goede balans te vinden in de ruisvermind0ering, doordat te veel ruisvermindering kan leiden tot een vlakkere afbeelding, waardoor de edges minder goed zichtbaar zullen zijn. De Gaussian ruisvermindering zal ook worden toegepast door middel van een 3x3 mask.

In Methode 3 wordt een high pass filter toegepast om de edge detection te implementeren. Hierbij wordt een geblurde versie van de originele afbeelding afgetrokken, waardoor alleen nog maar de sterke overgangen over blijven. Hierdoor worden alle hoge waardes doorgelaten, en alle lage waardes we gefilterd.

In Methode 4 wordt de sobel methode gebruikt. Sobel gebruikt en een verticale en een horizontale kernel om edges te detecteren. De nieuwe som van de absolute waarde die uit de verticale en horizontale filter komen vormen de gedetecteerde edges.

Ook wordt Thresholding gebruikt. Hierbij wordt een drempelwaarde gekozen op basis van het histogram na afloop van de bewerkingen. Deze thresholding zal worden geïmplementeerd door alle waarde onder de threshold zwart te maken, en alle waarden boven de threshold wit te maken.

Bij alle methode wordt dus threshoding toegepast. Het verschil tussen de verschillende methoden is het gebruik van ruisvermindering. Hierdoor kunnen we gaan vergelijken of ruisvermindering effect heeft op de kwaliteit van de edge detection.

## Keuze

Wij hebben nog geen keuze gemaakt voor een methode

## Implementatie

Je geeft aan hoe deze keuze is geïmplementeerd in de code

## Evaluatie

Je geeft aan welke experimenten er gedaan zullen worden om de implementatie te testen en te ‘bewijzen’ dat de implementatie daadwerkelijk correct werkt. Dit geeft direct informatie over de meetrapporten die er zullen worden gemaakt.

Om de implementatie te testen zal het masking systeem worden geïmplementeerd. Door dit te implementeren is het redelijk eenvoudig om ruisvermindering, de Laplacian en de high pass filter toe te passen op de afbeeldingen. Hierdoor kunnen we alle drie de methode implementeren en testen. We testen ze op de snelheid waarmee de edge detection zal worden uitgevoerd en de kwaliteit van de uiteindelijke edge detection.