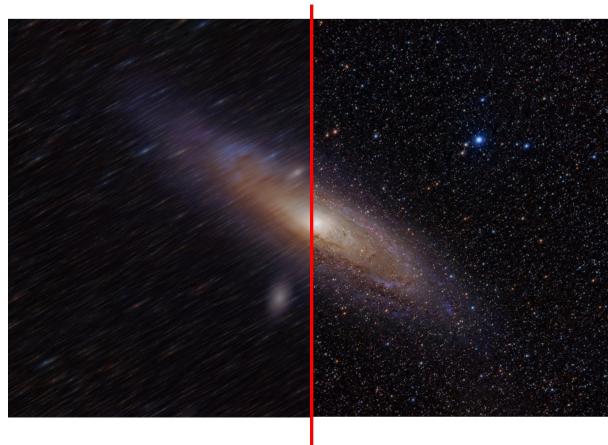


Labo 5: Beeldrestauratie

Gianni Allebosch, Martin Dimitrievski



Figuur 1. Afbeelding voor (links) en na (rechts) restauratie door middel van een Wiener filter.

1 Inleiding

Wanneer een beeld wordt vastgelegd of overgedragen, kan het onderhevig zijn aan verschillende soorten degradaties. Denk aan ruis, veroorzaakt door elektronische componenten in de camera of tijdens de overdracht, onscherpte door beweging van de camera of het object, of verlies van informatie door compressie. Beeldrestoratie heeft als doel om deze degradaties te corrigeren en de oorspronkelijke, onbeschadigde afbeelding zo goed mogelijk te reconstrueren.

In dit labo zullen we een aantal technieken en algoritmen voor beeldrestauratie bestuderen en toepassen, vooral gebaseerd op het *Wiener filter*. We beginnen met het begrijpen van de theorie achter beeldvorming en de verschillende soorten degradaties die kunnen optreden. We bouwen hierbij ook verder op de technieken die jullie aangeleerd kregen in de voorgaande labo's.

2 Theoretische achtergrond

In deze sectie herhalen we kort een aantal theoretische concepten aan rond beeldrestauratie. Voor meer uitgebreide informatie verwijzen we uiteraard naar de theorielessen, specifiek de presentatie rond beeldrestauratie (Les 6)

2.1 Basisprincipes beelddegradatie

Beelddegradatie verwijst naar alle processen die de kwaliteit van een beeld verminderen. Denk aan een foto die wazig is, ruis bevat of kleuren die vervaagd zijn. Deze degradaties kunnen optreden tijdens het vastleggen van het beeld (bijvoorbeeld door een bewegend object of slechte lichtomstandigheden), tijdens de overdracht (bijvoorbeeld door compressie) of tijdens het opslaan (bijvoorbeeld door beschadiging van het medium). Hieronder beschrijven we een vereenvoudigd model waarmee degradaties kunnen beschreven worden.

Stel dat de originele waarde van een afbeelding (bijvoorbeeld de grijswaarde of voor één kleurkanaal) formeel wordt uitgedrukt als de functie $f(x, y)$, waarbij (x, y) een bepaalde beeldlocatie is. Degradaties die bijvoorbeeld wazigheid introduceren, worden vaak gemodelleerd als de *convolutie* (of correlatie) van het originele signaal met een tweede functie $h(x, y)$, net zoals bij een beeldfilter. Zo'n degradatiefunctie wordt in die context soms ook de *point spread function* of puntspreidingsfunctie genoemd. Ze beschrijft namelijk hoe een beeldpunt lijkt te worden ‘uitgespreid’ in zijn omgeving.

De degradatiefunctie is vaak circulair of symmetrisch (bijvoorbeeld Gaussiaans). Dit is bijvoorbeeld zo bij een camera die uit focus staat. Een andere veelvoorkomende degradatie is ‘motion blur’, waarbij beeldobjecten of de camera zelf bewegen tijdens het maken van de foto (typisch bij een opname met lange sluitertijd).

2.2 Invers filter en Wiener filter

Als $h(x, y)$ de enige bron van degradatie is, kun je in principe het originele beeld perfect reconstrueren. Bij omzetting naar het frequentiedomein, kun je de componenten van de degradatie namelijk ‘wegdelen’ om het gerestaureerde signaal \hat{f} te bekomen:

$$g(x, y) = h(x, y) \circledast f(x, y) \quad (1)$$

$$\Leftrightarrow G(u, v) = H(u, v)F(u, v) \quad (2)$$

$$\Rightarrow \hat{F}(u, v) = \frac{G(u, v)}{H(u, v)} \quad (3)$$

Het gedegradeerde signaal g in het frequentiedomein (G) delen door de degradatiefunctie (H), heet *inverse filtering*. In de praktijk wordt deze techniek echter zelden gebruikt. Een eerste reden hiervoor is numerieke stabiliteit.

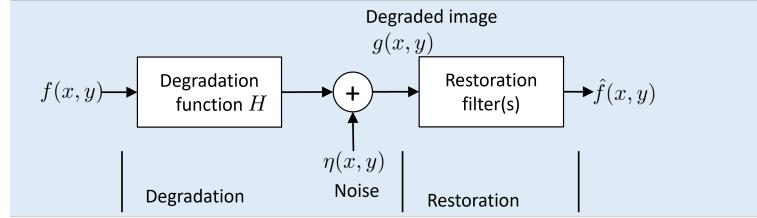
Vraag: Waarom? Wat is het gevaar bij een deling? **Hint:** denk aan wat er gebeurt bij frequentiecomponenten met een lage magnitude.

Een andere belangrijke reden is de mogelijke aanwezigheid van (additieve) beeldruis, los van de wazigheid. Een meer realistisch model van beelddegradatie (zie ook Figuur 2) is dus:

$$g(x, y) = h(x, y) \circledast f(x, y) + \eta(x, y) \quad (4)$$

$$\Leftrightarrow G(u, v) = H(u, v)F(u, v) + N(u, v) \quad (5)$$

$$(6)$$



Figuur 2. Model van beelddegradatie.

Het is duidelijk dat je de degradatie nu niet meer zomaar kunt wegdelen. Kleine waarden voor $H(u, v)$ zouden de ruisvariantie mogelijk erg versterken.

Om de bovenvermelde problemen te vermijden, werd het *Wiener filter* (of *Wiener deconvolutie*) uitgevonden. Dit filter is in staat om verschillende soorten beelddegradatie tegelijk aan te pakken: bijvoorbeeld door de wazigheid te verminderen, maar toch ook ruisonderdrukking uit te voeren. Het idee erachter is dat je sterkte van de ruisterm schat, en die informatie dan gebruikt om de restauratie uit te voeren. In zijn meest eenvoudige vorm kan dit filter beschreven worden met de formule

$$\hat{F}(u, v) = \frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + k} G(u, v) \quad (7)$$

$$= \frac{H^*(u, v)}{|H(u, v)|^2 + k} G(u, v), \quad (8)$$

waarbij H^* de complex toegevoegde van H voorstelt. k is hier een constante.

Vraag: Voor welke waarde van k is de werking van een Wiener filter dezelfde als die van een invers filter?

Vraag: In een meer geavanceerde vorm van het Wiener filter kan de constante k worden vervangen door een term die afhankelijk is van de frequenties (u en v). Wat is het voordeel hiervan (denk aan de verschillende soorten ruis)? Waarom is dit moeilijk toepasbaar in de praktijk?

3 Opdrachten

1. Programmeer een functie `Wiener_filter` die als inputparameters het originele beeld, de degradatiefunctie en de waarde voor k vraagt. Modelleer de degradatiefunctie als een Gaussiaan in het spatiale domein met standaardafwijking gelijk aan 3. (**Hint:** je kunt hierbij vertrekken van één van de functies uit labo3, waarbij je het filter dit keer ontwerpt in het spatiale domein.) Voer dit uit voor `Japan.blur.png` en toon het resultaat.
2. Voer nu het Wiener filter opnieuw uit voor de afbeelding `Japan_noisy_blur.png`, die meer (witte) ruis bevat. Hiervoor zul je de waarde van k moeten aanpassen. Zoek een passende waarde en toon het resultaat af. **Vraag:** Met welk

concept uit de signaalverwerking komt deze waarde gemiddeld gezien overeen, wanneer ze goed gekozen is voor een bepaald beeld? (Hint: we zoeken een specifieke verhouding.)

3. Gebruik het Wiener filter om het gedegradeerde beeld `Andromeda_motion_blur.png` te verbeteren. Om de degradatiefunctie te achterhalen, kun je volgende zaken gebruiken:
 - De meeste ‘objecten’ in de afbeelding zouden eigenlijk punten moeten voorstellen.
 - De degradatiefunctie verloopt in één bepaalde richting. (**Hint:** denk aan labo 4, hoe zou je deze richting kunnen bepalen?)
 - De hoeveelheid beeldruis is eerder klein.
 - OpenCV / NumPy hebben functies om lijnen te tekenen (`cv2.line()`), afbeeldingen te roteren (`cv2.getRotationMatrix2D()`, `cv2.warpAffine()`) en het bereik van afbeeldingen te clippen (`numpy.clip()`). Afhankelijk van jullie implementatie kunnen jullie deze mogelijk gebruiken.

4 Indienen

Indienen van het labo gebeurt via Opdrachten op Ufora. De code dien je in als 1 pythonscript genaamd **labo5.py** waarin alle opdrachten sequentieel worden uitgevoerd. Voeg een **README.txt** bestand toe waarin je programma beschreven is en waarin de vragen beantwoord worden. Alle gevraagde outputafbeeldingen geef je mee in een .zip-file genaamd **output.zip**.