

# Stripetykkelsesmåler

## Afsluttende Rapport For Praktikophold

Rudy Alex Kohn

24. Juli 2017

Rudy Alex Kohn  
ruak@force.dk  
s133235@student.dtu.dk



# Indeks

<b>Figurliste</b>	<b>3</b>
<b>Tabeller</b>	<b>3</b>
<b>1 Introduktion</b>	<b>4</b>
<b>2 Produktets specifikationer</b>	<b>5</b>
<b>3 Teknologivalg</b>	<b>6</b>
3.1 Kamera . . . . .	6
3.2 Laser . . . . .	6
3.3 OpenCV . . . . .	6
3.4 Begrundelse . . . . .	6
3.5 C++ . . . . .	6
3.6 Udviklingsværktøj . . . . .	6
3.7 PvAPI . . . . .	7
<b>4 Forsøgsopstillingen</b>	<b>8</b>
<b>5 Billedfelt - Opløsning</b>	<b>9</b>
5.1 Information . . . . .	9
5.2 Kortlægning af pixels . . . . .	9
<b>6 Processen</b>	<b>10</b>
6.1 Identificering af striben . . . . .	10
6.2 Lokalisering af laser . . . . .	10
6.2.1 På jorden . . . . .	10
6.2.2 På striben . . . . .	11
6.3 Begrænsninger . . . . .	11
<b>7 Føjlede forsøg</b>	<b>12</b>
7.1 Diagonal matricer . . . . .	12
7.1.1 Problemstilling . . . . .	12
7.1.2 Metode . . . . .	12
7.1.3 Resultat . . . . .	12
7.2 Differentiering . . . . .	12
7.2.1 Problemstilling . . . . .	12
7.2.2 Metode . . . . .	12
7.3 Resultat . . . . .	13
7.4 Histogram . . . . .	13
7.4.1 Problemstilling . . . . .	13
7.4.2 Metode . . . . .	13
7.4.3 Nødvendigheder . . . . .	13
7.4.4 Resultat . . . . .	13
7.5 Nærheds højdemåling . . . . .	14
7.5.1 Problemstilling . . . . .	14
7.5.2 Metode . . . . .	14
7.5.3 Resultat . . . . .	14

<b>8 Databehandlingen</b>	<b>15</b>
<b>9 Indledende undersøgelser</b>	<b>16</b>
<b>10 Nuværende resultater</b>	<b>16</b>
10.1 Antal billeder pr. måling . . . . .	16
10.2 Linearitet . . . . .	16
10.2.1 Verificering af kalibreringen . . . . .	16
10.2.2 Ny måling . . . . .	17
<b>11 Videre arbejde</b>	<b>18</b>
11.1 Bedre opstilling . . . . .	18
11.2 Sammenhæng af laser og kamera . . . . .	18
11.2.1 Fordel . . . . .	18
11.3 Kortlægning af real højde . . . . .	18
<b>12 Kommandolinje Interface</b>	<b>19</b>
12.1 Kommandoerne . . . . .	19
<b>13 Konklusion</b>	<b>20</b>
<b>Referencer</b>	<b>21</b>

## Figurliste

## Tabeller

1	Kommandolinje kontakter . . . . .	19
2	Kommandolinje: indstillinger . . . . .	19

## 1 Introduktion

Denne rapport afdækker praktikforløbet hos Delta (Force, Hørsholm). Den beskriver de forskellige faser i projektet, samt belyser de problematikker der opstod i forbindelse med at løse opgaven. Praktikopholdet hos Delta er baseret på et afklaringsprojekt med henblik på at få afdækket om hvorvidt det er muligt at kunne måle højden af en vejstribe (termoplast). I den forbindelse var der fra starten en del problematik, bl.a. var der på forhånd ikke udlagt hvorledes dette skulle foregå. På den baggrund startede forløbet med at der blev brugt tre uger på indsamling af information omkring hvilken form for teknologi der skulle benyttes. Valget endte med at stå mellem 3d laserskanning eller computer vision.

Det viste sig dog hurtigt at laserskanning var udelukket grundet de høje omkostninger det ville indebære, og der blev derfor fundet et ældre kamera frem der kunne benyttes til projektet.

## 2 Produktets specifikationer

Krav for at produktet rent faktisk kan blive en realitet.

- Processen må samlet set ikke tage særlig lang tid, gerne under 4 sekunder.
- Målingens præcision skal være på  $\pm 0.1$  mm
- Omkostningerne skal holde sig under 25.000,- DKr.

### 3 Teknologivalg

Denne sektion beskriver de forskellige delelementer projektet består af. Det er både med hensyn til hardware men hvilke udviklingsværktøj der blev benyttet. For at kunne danne et komplet billede, er der information medtaget omkring nogle af beslutninger og deres bevæggrunde.

#### 3.1 Kamera

Der er valgt at benytte sig af et kamera, der på baggrund af en laser der kaster en linje, kan måle forskel i højden fra hvor laserens linje kastes på en termoplastisk markering i forhold til, hvor den rammer ved siden af markeringen. Kameraet er monokromatisk for at udnytte lyset bedre og har en opløsning på 5 megapixel.

#### 3.2 Laser

Laseren kaster en linje ned på markeringsområdet med en styrke der gør det er muligt at kunne aflæse dens lokalisering. Forskellen på hvor laseren rammer på striben og hvor den ikke gør udgør den højde der er interessant.

#### 3.3 OpenCV

Til denne proces bliver Open Computer Vision [1] benyttet, hvilket er et gratis computer vision system der gør det muligt at lave behandling på dataen for at lokalisere laserlinjen på de billeder kameraet tager.

#### 3.4 Begrundelse

Valget af kamera og laser er baseret på prisen, et moderne kamera med en tilstrækkelig kvalitet koster under 6.000,- DKr, og en linjelaser koster mellem 2.000,- DKr. og 3.000,- DKr. En laser scannings enhed koster over 15.000,- Dkr. hvilket vil sætte for meget pres på budgettet.

Open Computer Vision er gratis, og kan benyttes i ubegrænset omfang, også i kommercielle forbindelser.

#### 3.5 C++

Udviklingen har indtil videre foregået i C++, da dette potentielt åbner op for at benytte sig af softwaren, eller dele af den, på generisk vis på tværs af diverse platforme, da softwaren er skrevet med udgangspunkt i standard template library. Softwaren kræver en compiler der understøtter C++14, for en komplet liste over understøttede compilere, se [her](#).

#### 3.6 Udviklingsværktøj

Softwaren er udviklet ved brug af Microsoft Visual Studio, Community Edition. Selve koden skulle være 100% kompatibel med andre udviklingskæder som f.eks. GCC. Det ville dog kræve at der blev sat et projekt op der sørger for de korrekte imports.

### **3.7 PvAPI**

Kameraet bliver kontrolleret ved hjælp af PvAPI. PvAPI er ellers blevet erstattet af det nyere API fra Allied Vision, Vimba. Men dette API kunne ikke bruges til projektet da det fejler med at kunne kontrollere elementære ting som f.eks. eksponering osv. Dog skal Vimba's drivere til kameraet været installeret på systemet.

## 4 Forsøgsopstillingen

Denne sektion omhandler opstillingen der er blevet benyttet til at lave test målinger med.



## 5 Billedfelt - Opløsning

Denne sektion omhandler billedfeltet kontra opløsning, og den problematik der ligger bag.

### 5.1 Information

Som udgangspunkt vil der i den afstand der skal tages et billede fra være kendt. Derfor er det muligt at udregne hvor mange pixels der er pr. mm (eller cm). Denne udregning betyder at hvis alle forvrængninger af billedet vil kunne blive taget med i den endelige udregning for en eventuelt højdeforskel baseret ud fra målingspunkterne.

### 5.2 Kortlægning af pixels

Ud fra de måleresultater der er fremkommet, er det muligt at kunne kortlægge de individuelle pixels til et antal mm. Ved en ru optælling af pixels kontra deres reale størrelse i millimeter, er der som udgangspunkt ca. 5 pixels pr. millimeter. Da dette er en utilstrækkelig opløsning for at kunne opretholde kravet omkring præcisionen (Jf. sektion 2 for oversigt), skal denne kortlægning foregå på subpixel niveau, ned til  $\frac{1}{100}$  pixel.

## 6 Processen

Denne sektion omhandler den process der ligger til grund for de resultater der foreligger.

### 6.1 Identificering af striben

Den hertil designede algoritme, vil forsøge at lokalisere striben der skal måles, ved at overtage kontrollen med kameraet og automatisk søge efter striben. Dette foregår ved at den justerer eksponeringen i trin, og undersøge de informationer der findes i et billedet der er taget efter hver justering. Årsagen til denne fremgangsmåde er, at det sikrer den for det bedst muligt udgangspunkt i senere stadier, da det eneste der er at forholde sig til er allerede kendte data og fordi der er et sammenhæng mellem de senere faser og den eksponering striben blev lokaliseret ved.

Stribens søgning er afhængig af forskellige parametre, hvor først og fremmest vinklen af striben i forhold til kameraet, ved aflæsningen, ikke må overstige 30 grader. Grundlaget for netop dette krav er, at der kan være for meget information der ikke er relevant hvis dette er tilfældet. Når det lykkedes for algoritmen at identificere noget der kunne være en stribe, bliver informationen verificeret mod sig selv for at sikre det ikke er åbenlyst irrelevante data. Kriteriet for at de fundne informationer bliver accepteret som værende tilhørende en stribe er, grundet den metode der bliver benyttet, at der kun må være to grupperinger af korrekt orienterede linjer, og at hver af linjerne i hver af samlingerne alle har skæringspunkter i hinanden. Der ud over bør grupperingerne være tilhørende hver deres halvdel af billedet<sup>1</sup>. Dette sikrer mod flyvske informationer fra elementer der kunne ligne kanten af striben bare en lille smule, som f.eks. et græsstrå eller andet der kunne befinde sig i nærheden.

Når denne del er færdig kan de informationer der er indsamlet blive brugt til at afskille de forskellige lokationer for hvor laseren befinder sig, både på striben og ved siden af den.

### 6.2 Lokalisering af laser

Denne sektion omhandler de to faser der lokaliserer laseren i billedet. Både der hvor den er på striben og hvor den er på jorden.

#### 6.2.1 På jorden

Ud fra den information der er indsamlet omkring sribens lokalitet, kan laserens lokation udenfor striben nemmere findes. Algoritmen drager nytte af at laseren altid vil befinde sig på den nederste fjerdedel af billedet, og kan derfor udelukke en del fra start. For at laseren kan lokaliseret og at den rent faktisk er brugbar, skal den være "rimelig" vandret, dvs. den må f.eks. ikke være for bøjet. Eksponeringen fordobles ud fra den eksponering benyttet til at lokalisere striben, da området på forhånd forventes at være mørkere da der ikke er nogen stribe. Laseren findes ved først at massere data for at gøre det nemmere at lokalisere vandrette former, og derefter at lede efter linjer der har under en grads hældning. Hvis dette lykkedes bliver der fokuseret på området hvor der er fundet linjer, da dette er med størst sandsynlighed laserens placering i billedet. Grundet at laseren er sporadisk mht. dens afbildning, bliver laserens lokation i højden udregning efter dens massemidtpunkt, hvor dens masse rent faktisk er dens intensitet pr. kolonne. Dette sikrer en så nøjagtig lokalisering som muligt.

---

<sup>1</sup>Sikrer en mere stabil søgning af stribe under de nuværende testforhold.

### 6.2.2 På striben

Da sribens lokation allerede er kendt, skal laseren nu lokaliseres oven på striben. Da der i første fase blev benyttet en så lav som mulig eksponering, betyder dette at laseren har den kraftigste intensitet i dette område. Denne information kan udnyttes til at frasortere alt andet og derved er laserens lokation fundet. Herefter er fremgangsmåden den samme som med laseren på jorden, hvorved gennemsnittet over  $N$  antal billeder bliver trukket fra det samme form for gennemsnit fra laseren på jorden. Derved er forskellen i højden på de to steder fundet.

## 6.3 Begrænsninger

Opsætningen gør, at der er en del begrænsninger.

- Ustabilitet, forholdet mellem kamera og laser ændrer sig meget nemt
- Laseren buer. Over 400mm har den en bøjning på op til 2mm
- Høj ambient lysstyrke
- Vinklet måling. Ikke relevant grundet opsætningens natur ikke kan håndtere dette, da både laser og kamera skal være  $< 0.1\text{mm}$  synkront positioneret hele målingen i gennem

## 7 Fejlede forsøg

Undervejs i forløbet er forskellige metoder til stadierne i processen blevet afprøvet, hvoraf alle endte med at blive skrottet. Denne sektion beskriver de forskellige procedurer, hvilket problem de skulle løse og hvad ideen bag dem var, og hvorfor de ikke var brugbare i praksis.

### 7.1 Diagonal matricer

#### 7.1.1 Problemstilling

Identificering af start og slutpunkter for laseren på striben

#### 7.1.2 Metode

Opdele billedet i diagonale matricer for derefter at gennemløbe disse for at identificere dobbeltskæringer af laseren. Selve billedet ville opdelt i matricer, præcist det dobbelt antal i forhold til billedets højde, på begge leder. Dette ville resultere i et net af matricer på tværs i begge retninger, hvor pixel intensitetsforskelle for hver i forhold til dens naboer ville medføre en slags toppunkt. I tilfælde af at to var tilstede og det matchede med naboerne, kunne denne information benyttes til at søge efter intensitets variationer, der i sidste ende ville medføre at en potentiel lokation der kunne bruges var fundet.

#### 7.1.3 Resultat

Denne metode viste sig at være utroligt krævende, både med hensyn til design og med hensyn til regnekraft samt noget utilregnelig. Under perfekte situationer virkede det nogenlunde, og hvis der var nogen diskrepans i laseren position grundet støj eller andre elementer ville den give resultater der ikke kunne bruges. Den blev derfor kasseret i et forholdsvis tidligt stadie da det ikke var rentabelt at gå videre i denne retning.

### 7.2 Differentiering

#### 7.2.1 Problemstilling

Lokalisering af laser på stribe kontra jord.

#### 7.2.2 Metode

Ved at tage alle punkter i X-aksen, og tage forskellen på dem i stigende orden, ville det være muligt at se om et givent koordinat steg eller faldt i højde. Denne process udføres to gange, for at få absolutte værdier der indikerer udsving i niveau. Ved at finde de største af disse var forhåbningen af det kunne bruges til at finde ud af hvor den stigning laseren var på striben befandt sig.

### 7.3 Resultat

I teorien var det en fantastisk ide, men det forblev også kun i teorien. Det viste sig at grundet afveksling af laserens intensitet på jorden, f.eks. fra fremmedlegemer såsom småsten eller deres refleksioner herfra, hurtigt kunne gøre det af med den metode. Det viste sig derfor umuligt at være sikker på om de punkter der blev lokaliseret rent faktisk var stribens position eller noget der bare havde dannet en afvigelse et helt andet sted. På trods af det var det dog en ganske effektiv og simpel metode til at identificere niveauvariationer.

### 7.4 Histogram

#### 7.4.1 Problemstilling

Lokalisering af stribe

#### 7.4.2 Metode

Ved at opdele alle billedets pixels i et intensitets histogram kunne de forskellige intensiteter være med til at identificere hvilke der tilhørte striben og derved lokalisere dens position.

#### 7.4.3 Nødvendigheder

Algoritmer til at identificere højde- og lavpunkter ud fra data samt en manuel opbygget histogram datastruktur.

#### 7.4.4 Resultat

I teorien skulle det være muligt, men her opstod et problem der ikke var forudset. Intensiteten af pixels for

- Stribe med og uden laser
- Jord med og uden laser

befandt sig inden for et så lille område at det praktisk var umuligt at foretage udvælgelse af værdierne for videre identificering. Det viste sig også at der, på trods af udelukkelse af for mørke dele, ikke var mulighed for at garantere hvilke intensiteter der tilhørte hvad. Dette forsagede desværre også at det ikke var muligt at søge efter intensitetsgrænser. Derudover blev situationen heller ikke forbedret grundet eksponeringsproblematikker og støv på kamerachippen osv. Det havde den effekt at det var stort set umuligt at separere de forskellige elementer da overgangene var utroligt ustabile og kunne blive influeret af hvad som helst. På den lyse side blev der dog udviklet to algoritmer, en til at finde højdepunkter og en til at finde lavpunkter, der begge er justerbare i med forskellige parametre.

## **7.5 Nærheds højdemåling**

### **7.5.1 Problemstilling**

Undgå indflydelse fra skæv laser ved måling af højden.

### **7.5.2 Metode**

Grundet laserens bueform, blev to små bider af laserens udskåret, hvis disse faldt inde for samme vinkel ville betydningen af laserens bueform have minimal indflydelse. Desværre er denne metode ikke præcis nok da den ikke tager højde for bl.a. udsving i laserens intensitetsvariationer over tid eller elementer der kunne være betydningsfulde for målingen som f.eks. fremmedlegemer.

### **7.5.3 Resultat**

Målinger foretaget ved denne metode viser markant andre værdier, og er temmelig svingene i resultater.

## 8 Databehandlingen

## 9 Indledende undersøgelser

De første tre uger blev stort set kun brugt på at søge efter information omkring hvordan det var muligt at skulle kunne få noget information der kunne bruges til noget. Der blev primært kigget efter 3d laser scannings enheder, da det ville være det mest præcise og kræve mindst i opsætning og udvikling osv. På trods af at budgettet var på 25.000,- DKR, blev det i sidste ende vurderet for dyrt med en laser scannings enhed til omkring 18.000,-. Det virkede i det hele taget ikke som om der var nogen der havde skænket projektet en tanke på noget område før det startede. Derfor blev der fundet et næsten 10 år gammelt kamera frem der kunne bruges, og der blev bestilt en laser i stor hast der kunne bruges til formålet. 4. uge kom der så gang i udviklingen, hvilket betød at ud over de alm. børnesygdomme grundet konfigurationen af kameraet, også skulle bruges et API til kameraet. Desværre viste det sig at deres API ikke virkede når det kom til at få informationer fra kameraet, som f.eks. dets indstillinger og/eller billeder. Derefter gik der så en uge med at få sat en OpenCV op til at benytte det ældre API, konfigurere det hele og få hul igennem til kameraet via C++.

## 10 Nuværende resultater

Denne sektion beskriver de resultater, både de endelige samt metoder og algoritmer brugt og udviklet under forløbet.

### 10.1 Antal billeder pr. måling

Grundet fluktuerende intensitet omgivet laseren, er det nødvendig at udføre hver enkelt måling over en billedserie.

### 10.2 Linearitet

Ud fra en række målinger i serie, dvs. i højder der spænder fra 1mm til 6mm, er der blevet opsat en test model for hvorvidt der er linearitet.

(indsæt tabel her)

Den ovenstående tabel viser de målinger der er foretaget i forskellige højder. For at sikre et rimeligt spænd er hver enkelt måling bestående af 25 målinger, hvor de hver er foretaget over 25 billeder.

Ud fra de ovenstående tal, viser det sig at der rent faktisk er et rimelig god linearitet i målingerne. Dette afgøres ved  $R^2$  værdien der indikerer lineariteten, og en værdi på 1.00 er perfekt og vil kun opstå ved fuldstændig uniform data.

Disse målinger udgør kalibreringen, dvs. reference.



### 10.2.1 Verificering af kalibreringen

Ved at regne baglæns med de givne informationer, er det muligt at lokalisere anormaliteter i de enkelte højder. Resultatet angiver hvor præcis lineariteten rent faktisk er, da resultaterne gerne skulle svare til højden i millimeter, der i dette tilfælde er.

Den angivne linjeformel er

$$px2mm = \frac{avg - b}{a} \quad (1)$$

Indsætning af de resulterede data i denne formel giver følgende

(data her)

Diskrepansen for selve målingerne findes ved (i promille)

$$dev\%_0 = |px2mm - mm| \cdot 1000 \quad (2)$$

(formel her)

Dette frembringer følgende resultater, hvor det er tydeligt de ligger ret tæt på selve målingerne.

### 10.2.2 Ny måling

Verificering af kalibreringsmålingen er som udgangspunkt kun god til at tjekke om de informationer man har til rådighed er fornuftige.

25 nye målinger ser således ud

Ved indsætning i kalibrerings formlerne fås således

(indsæt resultat her)

## 11 Videre arbejde

Denne sektion afdækker områder hvorpå der kan arbejdes videre med i projektet.

### 11.1 Bedre opstilling

Den nuværende opstilling er primitiv, dette gør den er meget usikker og følsom. Forbedringer på dette punkt vil være en stor fordel. Hvorledes dette realiseres, er afhængig af hvilken retning projektet drejer, og ligger på nuværende tidspunkt kun grund til antagelser bedst udeladt.

### 11.2 Sammenhæng af laser og kamera

De to hovedelementer i opstillingen, kameraet og laseren, er separeret på nuværende tidspunkt. Dette gør det meget svært at lave vinklede målinger, da flytningen af den ene kræver at den anden flyttes i samme grad, hvilket er et stort problem at gøre med den påkrævede nøjagtighed individuelt.

#### 11.2.1 Fordel

At kunne lave målinger hvor kamera er vinklet i forhold til striben der skal måles højde på. Da både laser og kamera ville flytte sig sammen.

### 11.3 Kortlægning af real højde

Det foreslås der udregnes alle kortlægninger på forhånd så de kan benyttes til at få den reale position i billedet i millimeter, ud fra pixel lokationer. Jf. sektion 5.2 for præliminær information omkring kortlægning.

## 12 Kommandolinje Interface

Softwaren understøtter styring via et hertil designet kommandolinje interface. Denne sektion beskriver de forskellige muligheder stillet til rådighed ud fra det.

Fungerer ved at starte programmet med de valgte parametre, som f.eks.

```
ThicknessGauge.exe -d --glob_name=camera --show_windows=0
```

For en liste over de muligheder der er til rådighed, kan programmet startes med:

```
ThicknessGauge.exe --help
```

### 12.1 Kommandoerne

\* denoterer ikke komplet understøttet, eller ikke komplet i softwarens nuværende tilstand.

-d	--demo	Kører softwaren i dens regulære tilstand
-c	--calibrate	Kører kamera kalibrering
-t*	--test*	Kører test tilstand
-g*	--save_glob*	Kører en prædefineret opsætning og gemmer en billedserie.

Tabel 1: Kommandolinje kontakter

--phase_two_exposure*	Tvinger programmet til at bruge en eksponering i fase to.
-z*	Tvinger aflæsning af område, til nulmålinger.
--glob_name	Brug gemt billedserie, eller "camera" for live feed.
--opencv_threads	Tvinger OpenCV til at bruge N antal tråde.
--calibrate_output	Filnavn til at gemme kamera kalibrering under.
--camera_settings	Kamera kalibrerings fil der skal bruges under processen.
--record_video*	Opretter en video fil af processen.
--show_windows	Optegner processen visuelt. (Giver ikke nøjagtige målinger.)
--frames*	Sætter antal frames der skal benyttes under kalkulationerne.
--test_suite*	Filnavn for at gemme test sæt under.
--version	Udskriver programmets version.
--help	Viser komplet hjælp af alle kommandolinje parametre.

Tabel 2: Kommandolinje: indstillinger

## 13 Konklusion

Målet var at få afklaret om det kan lade sig gøre at lave et produkt der kan måle tykkelsen af en vejstribe. På trods af en del problematikker omkring testopsætningen, er det lykkedes at lave nøjagtige målinger med en god præcision. Derfor mener jeg at hvis forholdene blev forbedret, at kvaliteten af målingerne vil øges, på trods af at de allerede på nuværende tidspunkt måler med langt større præcision end kravet. Hvis det kan lade sig gøre under så basale og problemfyldte omstændigheder, så kan det bestemt blive et godt produkt hvis det bliver sat ordentligt sammen.

Der er derfor ingen grund til at projektet ikke kan betegnes som en succes.

## Referencer

- [1] OpenCV. OpenCV (Open Source Computer Vision) er et bibliotek der indeholder programmeringsfunktioner for 'real time computer vision'. Information tilgængelig på <http://opencv.org/>.