Kravspecifikation

Fredrik Berntsson

Version 2.0

Status

Granskad	FB	2025-01-21
Godkänd	FB	2025-01-21

Dokumenthistorik

Version	Datum	Utförda ändringar	Utförda av	Granskad
1.0	2014-01-15	Första versionen	FB	FB
1.8	2022-01-13	Uppdaterad för VT-2022	FB	FB
2.0	2025-01-21	Uppdaterad för VT-2025	FB	FB

1 Introduktion

Manipulation av bilder är en vanligt förekommande tillämpning där matematiska metoder används för att lösa ett flertal problem som exempelvis att ändra upplösning eller ändra detaljer i bilden. En annan tillämpning är datakompression där bara den väsentliga informationen i bilden lagras för att spara minne. Dessa tillämpningar kan beskrivas som *interpolations problem*. Vi skall inom projektet formulera interpolationsproblemet i 1D på ett sådant sätt att det går att generalisera till 2D (eller till 3D vilket i detta fallet skulle tolkas som en filmsekvens). Vi skall dessutom visa hur exempelvis bildkompression kan formuleras som ett interpolationsproblem. Slutligen skall ett system implementeras som enkelt låter oss lösa olika tillämpningsproblem. Systemet skall utformas på ett sådant sätt att det är flexibelt och lätt att modifiera. Detta för att vi skall kunna utveckla systemet genom att exempelvis implementera mera avancerade metoder för interpolation.

1.1 Parter

Med leverantör åsyftas i efterföljande text projektgruppen. Matematiska Institutionen (MAI) är beställare och kund.

1.2 Syfte och Mål

Målet med projektarbetet är att inhämta kunskap i hur matematiska metoder kan användas i realistiska tillämpningar, samt inhämta tillräcklig kunskap inom tillämpningen bildbehandling för att förstå vilka möjligheter och begränsningar som metoderna som studeras har. Projektet har också som övergripande mål att ge träning i att utveckla ett system, ingenjörsmässiga arbetsmetoder, samt dokumentation och muntlig presentation.

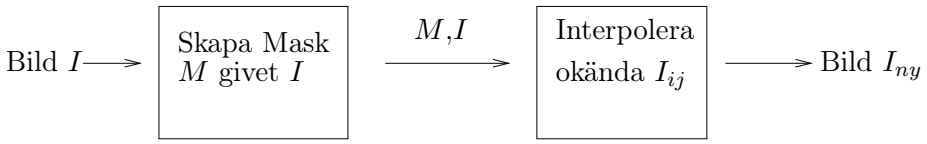
1.3 Användning

Beställaren skall kunna använda systemet för att lösa ett antal olika bildbehandlingsuppgifter. Systemet skall utformas så att det är enkelt för användaren att vidareutveckla.

2 Systemöversikt

Kärnan i systemet skall vara en funktion som löser ett interpolationsproblem. Indata är en *bild*, dvs en matris $I = (I_{ij})$ där en enskild pixel antigen har ett

känt värde eller betraktas som okänd samt en *Mask*, dvs en matris $M = (M_{ij})$ som talar om vilka pixlar som är kända respektive okända. För att lösa ett bildbehandlingsproblem genomförs görs steg enligt Figur **??**.
De olika bildbehandlingsproblemen svarar mot olika metoder för att konstruera mask martisen.



Figur 1: Bildbehandlingsaproblem löses genom att man först skapar en mask M som talar om vilka platser i I matrisen som skall betraktas som okända. I interpolationssteget beräknas lämpliga värden på de okända pixlarna I_{ij}.

2.1 Avgränsningar

Systemet skall ses som en prototyp och inget högre krav ställs på användargränssnitt. Det räcker om systemet består av ett antal fristående funktioner som tillsammans kan användas för att lösa uppgiften. Det måste isåfall medfölja ett tydligt exempel där det visas hur de ingående funktionerna skall användas tillsammans för att lösa bildbehandlingsproblemen.

2.2 Generella krav på hela systemet

Krav 1	**Original**	Systemet skall kunna lösa bildbehandlingsproblem	**Bas**
Krav 2	**Original**	Systemet skall vara testat och stabilt med avseende på parametrar användaren kan välja.	**Bas**

3 Interpolationsproblemet

Vi börjar med att beskriva en metod för linjär interpolation i en dimension. Antag att vi har en vektor y, med element y_i, där vissa element är kända och vissa okända. Inför vi en x vektor av samma längd som y, med exempelvis $x_i = i$. Om då y_j och y_k är kända värden men y_i för $j < i < k$ är okända kan vi hitta den räta linjen som går genom punkterna (x_j, y_j) och (x_k, y_k) och använda den för att approximera de okända värdena. För att enkelt hålla reda på vilka punkter som är kända respektive okända använder vi en vektor

m sådan att $m_i = 1$ om elementet y_i är att betrakta som känt och $m_i = 0$ om y_i är okänt.

Krav 3	**Original**	Visa teoretiskt hur interpolationsproblemet i $1D$ kan beskrivas som ett randvärdesproblem för en ordinärdifferentialekvation.	**Bas(1)**
Krav 4	**Original**	Implementera linjär interpolation i $1D$ enligt den metod som beskrivs i Krav 1.	**Bas**
Krav 5	**Original**	Skapa minst ett testexempel som visar att metoden fungerar	**Bas**

Då vi istället studerar interpolation i $2D$ eller $3D$ behöver vi en formulering av interpolationsproblemet som är enkel att generalisera. Krav 3 erbjuder en sådan formulering. En bild I kan ses som en funktion $I(x, y)$, , $1 \leq x \leq n$, $1 \leq y \leq m$, där $n \times m$ är antalet pixlar i bilden. Vissa pixlar betraktas som kända och vissa som okända. Vi kan beskriva detta genom att införa ett område Ω sådant att I_{ij} är okänd ifall $(x_i, y_j) \in \Omega$ och känd annars. Området Ω kan på ett enkelt sätt beskrivas av en matris M, eller *Mask*, sådan att $M_{ij} = 1$ om I_{ij} är att betrakta som känd, och $M_{ij} = 0$ annars.
Då vi formulerat interpolationsproblemet som ett randvärdesproblem för en partiell differential ekvation kan lösa problemet med hjälp av en finita differensapproximation. Detta leder till ett linjärt ekvationssystem där de obekanta är pixelvärdena I_{ij}.

Krav 6	**Original**	Visa hur formuleringen i Krav 3 kan generaliseras till $2D$. Beskriv tydligt hur interpolationsproblemet kan formuleras som ett randvärdesproblem i planet.	**Bas(1)**
Krav 7	**Original**	Välj en finita differensapproximation av metoden i Krav 6 och visa hur det linjära ekvationssystemet ser ut. Redovisa här eventuella restriktioner på bilden för att metoden skall fungera.	**Bas(1)**
Krav 8	**Original**	Implementera interpolation i $2D$ enligt den metod som beskrivits i Krav 6 och Krav 7.	**Bas**
Krav 9	**Original**	Hitta ett bra test exempel som tydligt visar att metoden fungerar. En analytisk lösning på problemet som besrivits i Krav 6 skall användas.	**Bas**
Krav 10	**Extra**	Om finita differensapproximationen implementeras korrekt skall felet bero på antalet gridpunkter som används, eller på steglängden $h = \Delta x = \Delta y$. Genomför en studie som visar att felet jämfört med en analytisk lösning kan beskrivas som Ch^p. Hitta även p experimentellt.	**Bas**

För varje konkret bild I måste vi hitta ett effektivt sätt att beskriva det område där pixlarna I_{ij} skall betraktas som okända. I Matlab finns en funktion `ginput` där man kan klicka i ett grafikfönster där en bild visas och få koordinaterna man klickade på. Detta kan utnyttjas för att definiera rektangulära områden genom att man klickar på två av dess hörn. Vi kan även kombinera två olika områden (eller Masker) M_1 och M_2 genom logiska villkor. Följande möjligheter skall finnas

- Visa orginal bilden i ett grafik fönster och markera ett rektangelområde som skall betraktas som okänt. Skapa motsvarande mask matris.
- Kombinera två mask matriser till en på ett sådant sätt att en pixel betraktas som okänd om den är okänd i någon av de ingående mask matriserna.
- Visa orginalbilden i ett grafikfönster och rita in kanten på det okända området.

Ett problem är att för stora bilder blir antalet obekanta, och därmed matrisstorleken, mycket stort. Detta kan ställa till problem då systemet $Ax = b$ löses på grund av att tillräckligt med minne saknas på datorn. Delvis kan problemen reduceras genom att använda ett *glest lagringsformat* men vi kan även observera att flera rader i matrisen endast innehåller ekvationen $1x_i = b_i$, utan att samma obekanta x_i ingår i någon annan ekvation. Sådana obekanta kan strykas från ekvationssystemet och man måste då stryka både motsvarande rad och kolumn från matrisen. För att hålla reda på vilka obekanta som skall ingå i ekvationssystemet skapar man en index vektor `ind` som innehåller alla index i för vilka variabeln x_i skall ingå i det reducerade ekvationssystsmet. Givet det ursprungliga ekvationssystsmet $Ax = b$ beräknar vi då först det reducerade ekvationssystsmet $\bar{A}\bar{x} = \bar{b}$ och vektorn `ind`. Det reducerade systemet löses och från lösningen $\bar{x}$, högerledet b, och vektorn `ind` kan vi sedan återskapa lösningen till det fulla ekvationssystemet x.
Det går givetvis att skapa det reducerade ekvationssystemet direkt utan att ta omvägen att först skapa det fulla $Ax = b$ men det är oftast praktiskt att börja med det enklaste fallet och sedan successivt utveckla metoden ett steg i taget.

Krav 11	**Original**	Implementera lämpliga funktioner för att skapa och manipulera mask matriser enligt ovan.	**Bas**
Krav 12	**Original**	Skapa en mask matris som svarar mot ett cirkulärt område. Klicka i cirkelns centrum samt på en punkt på cirkeln.	**Extra**
Krav 13	**Original**	Implementera metoden att reducera storleken på ekvationssystemet enligt ovan.	**Extra**

4 Tillämpningar

Interpolation i bilder (dvs $2D$) har flera tillämpningar. Vi skall nu försöka beskriva åtminstone några av dessa. Den enklaste tillämpningen är då en bild innehåller skadade områden, eller oönskade detaljer. Isåfall kan man skapa en mask matris som beskriver vilka delar av bilden man skall ersätta med nya pixelvärden. Vi använder sedan interpolation i $2D$ för att beräkna nya pixelvärden som ersätter de existerande. Vill man på detta sätt manipulera färgbilder så betraktar man färgbilden som tre separata bilder (röd,grön,blå).
Nästa tillämpning är att ändra upplösning på bilder, eller så kallad digital zoom. Givet en $n \times m$ bild I skapar vi en ny dubbelt så stor bild $\tilde{I}$ genom

att betrakta pixlar med jämnt index, $\tilde{I}_{2i,2j} = I_{i,j}$, som kända och pixlar med udda index som okända. Detta fungerar ej för pixlar som ligger på kanten av bilden så där måste först endimensionell interpolation användas.
Den metod vi använder för att interpolera innebär att vi löser en differential ekvation och approximerar bilden I. Den exakta bilden $I(x, y)$ kommer säkert inte att vara en lösning till differentialekvationen i hela området. Dock innehåller ofta bilder stora områden där färgerna är konstanta eller långsamt varierande. I sådana områden kan bilden återskapas väldigt noggrannt med interpolation. Detta betyder att vi kan minska mängden data vi måste lagra genom att evaluera en lämplig differntial operator för varje pixel (x_i, y_j). Om resultatet till beloppet är mindre än en tolerans `tol` så kan vi betrakta den punkten som okänd och istället beräkna pixelvärdet genom interpolation. Vi behöver då endast larga några pixlar där viktig information finns. Detta kallas kant-baserad bildkomprimering. Hur stor tolerans `tol` vi skall använda beror på vilken kvalitet vi kräver av den återskapade bilden. Givet att masken är M kan vi i Matlab använda `spy(sparse(M))` för att grafiskt se vilka pixlar som måste användas för att kunna återskapa bilden noggrannt via interpolation.
För att mäta hur nära den återskapade bilden är den ursprungliga kan Frobenius norm användas. Vi mäter då skillnaden som

$$\|I - \tilde{I}\|_F = \sqrt{\sum_{i,j} |I_{ij} - \tilde{I}_{ij}|^2}.$$

Krav 14	**Original**	Använd interpolation i $2D$ för att ta bort oönskade detaljer i en färgbild. Bild tillhandahålls av beställaren.	**Bas**
Krav 15	**Original**	Implementera en funktion som använder interpolation för att dubbla upplösningen på en bild.	**Extra**
Krav 16	**Original**	Implementera en funktion som genomför bildkomprimering enligt ovan. Givet en bild skall en mask som beskriver de pixlar vi skall betrakta som kända respektive okända returneras.	**Bas**
Krav 17	**Original**	Gör ett tydligt exempel där kant-baserad datakomprimering används för att minska antalet pixlar vi behöver larga.	**Bas**
Krav 18	**Original**	Gör en studie som visar hur den valda toleransen `tol` vid bildkomprimering bestämmer antalet pixlar vi måste lagra samt kvaliteten på resultatet mätt i Frobenius norm.	**Extra**

5 Ekonomi

Krav 17	**Original**	Projektet skall genomföras med en arbetsinsats på 105 timmar per student.	**Bas***

6 Leveranser

Vid slutleverans skall samtliga krav märkta **Bas** vara uppfyllda. För grupper med 5 studeneter gäller att minst ett krav märkt **Extra** dessutom skall vara uppfyllt. För grupper med 6 studenter gäller att tre **Extra** krav skall vara genomförda. För större grupper gäller att samtliga **Extra** krav skall vara uppfyllda.

Krav 18	**Original**	Leverans av gruppkontrakt skall ske till handledare och kursansvarig.	**Bas**
Krav 19	**Original**	Leverans av projektplan skall ske via e-post till beställaren och handledaren.	**Bas**
Krav 20	**Original**	Presentation av krav Bas(1) skall ske muntligt för beställare. Detta sammanfaller med BP3.	**Bas**
Krav 21	**Original**	Delleverans omfattande det utvecklade systemet, bruksanvisning, och ett utkast till den konferensartikel som presenterar arbetet skall skickas till beställare och handledare.	**Bas**
Krav 22	**Original**	Slutleverans bestående av den artikel som beskriver gruppens arbete skall skickas till handledare och beställare. Det skall tydligt framgå i texten att alla Bas-krav skall är uppfyllda samt extrakrav enligt texten ovan.	**Bas**
Krav 23	**Original**	Statusrapporter skall skickas via e-post till beställare varannan måndag under tiden som projektarbetet pågår.	**Bas**
Krav 24	**Original**	Tidsrapporter för varje vecka skall skickas via e-post till handledare och beställare senast måndag 13.00 påföljande vecka	**Bas**
Krav 25	**Original**	Leverans av efterstudie skall ske till kursansvarig.	**Bas**
Krav 26	**Original**	Krav märkta Bas* skall vara uppfyllda vid kursens avslutande och redovisning av detta sker genom mail till beställaren.	**Bas**

7 Dokumentation

För att beställaren skall kunna testa det utvecklade systemet behövs en bruksanvisning. Denna bruksanvisning skall förklara hur man gör för att använda systemets olika funktioner.

Krav 27	**Original**	En bruksanvisning skall följa med systemet	**Bas**
Krav 28	**Original**	En projektplan skall upprättas	**Bas**
Krav 29	**Original**	Mötesprotokoll skall föras vid alla möten	**Bas***
Krav 30	**Original**	Tidsrapporter skall sammanställas varje vecka	**Bas***
Krav 31	**Original**	Projektgruppensarbete skall beskrivas i en artikel som lämpar sig för att presenteras på en teknisk konferens. Artikeln skall innehålla nödvändig bakgrundsinformation, en beskrivning av vad som gjorts, samt exempel som illustrerar hur metoden fungerar. Detta dokument behöver inte följa LIPS mallarna.	**Bas***
Krav 32	**Original**	En litteraturstudie skall göras och någon, eller några, differential ekvationer som används för att interpolera bilder skall presenteras. Detta för att beställaren skall kunna vidare utveckla projektet.	**Bas***
Krav 33	**Original**	En efterstudie skall skrivas.	**Bas***
Krav 34	**Original**	Det skall skrivas en text om hur det utvecklade systemet kan bidra till ett hållbart samhälle. Omfattningen bör vara en till två paragrafer. Denna skall ingå i artiklens introduktion eller i sammanfattningen.	**Bas***
Krav 35	**Original**	Alla dokument som lämna till beställare skall granskas, med avseende på både språk och innehåll, av minst en projektdeltagare, utöver den som skrev texten.	**Bas***
Krav 36	**Original**	Alla dokument skall följa LIPS-mallarna	**Bas***

8 Utbildning

Inom projektet erbjuds utbildning i Matlab.