

NTNU
Norges teknisk-naturvitenskapelige
universitet

Fakultet for informasjonsteknologi,
matematikk og elektroteknikk

Institutt for datateknikk
og informasjonsvitenskap



**EKSAMEN I EMNE
TDT4195/SIF8043 BILDETEKNIKK
ONSDAG 19. MAI 2004**

LØSNINGSFORSLAG

OPPGAVE 1 Bildebehandling – Grunnleggende (100 poeng)

a) Et bilde som består av 1024×1024 piksler som kan representere 1024 grånivå, skal overføres på en seriell linje med hastighet 9600 baud. Pikslene må bestå av hele byte, og det anvendes ingen form for kompresjon:

1. Hvor mange byte må overføres? (10 poeng)

Each pixel occupies 2 bytes. The number of bytes in the image is $1024 \times 1024 \times 2 = 2097152$ or 2 Mbytes.

2. Hvor lang tid vil overføringen ta? (10 poeng)

The transmission rate of 9600 baud, with no compression scheme in use, uses 10 bits (start bit + 8 data + stop bit) giving 960 data bytes per second.

The time taken is $1024 \times 1024 \times 2 / 960$ seconds = 2184.54 seconds.

Bearing in mind the many students did not know what a baud was, and they will probably not think of start and stop bits we should probably allow $1024 \times 1024 \times 2 \times 8 / 9600 = 1747.62$ seconds as an alternative.

Confusion of baud with bytes/sec or kBytes/sec should give one half credit.

3. Foreslå kortfattet en applikasjon hvor 1024 grånivå kan være mer nyttig enn de vanlige 256. (10 poeng)

Range images using 8 bit pixels are too imprecise for good surface fitting. Range images with 1024 levels are much more useful in building a 3D model of the environment.

The illumination of a solid object generally produces a range of intensities that is better represented in 1024 grey levels than 256. Automated image analysis will benefit from the greater dynamic range.

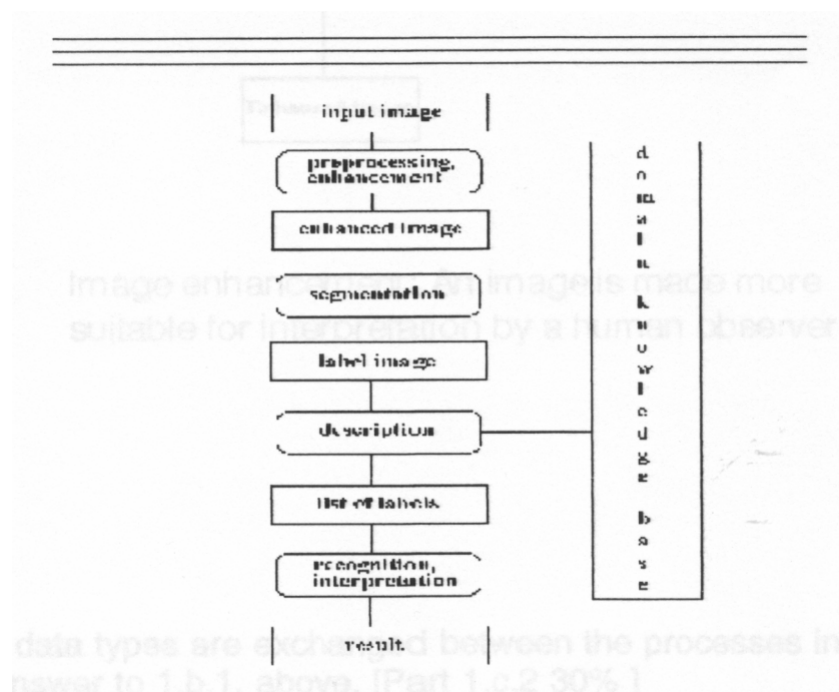
Other suggestions, eg. presentation of X ray data to a human could be given for half credit.

The paradox is that human vision does not simultaneously resolve more than a few different intensities in a single image. The human appears to limit the area of interest in an image by adjusting viewing distance so that, overall, a better appreciation of the content is obtained by remembering the content in adjacent patches.

- b) Tegn to merkede diagram som viser de nødvendige bildebehandlingskrittene for henholdsvis:

1. automatisk mønstergjenkjenning og (20 poeng)

Computer Vision Research - Introduction.



Automated analysis: An image is processed for interpretation by an algorithm.

2. presentasjon for en menneskelig observatør (20 poeng)

Computer Vision Research – Introduction

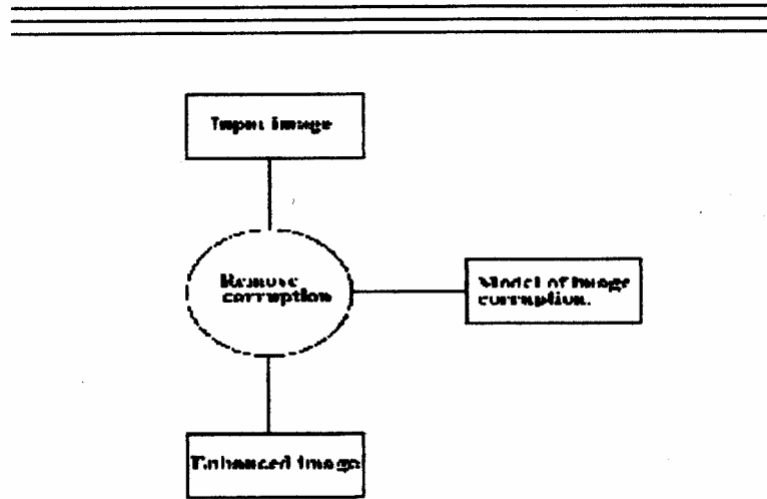


Image enhancement: An image is made more suitable for interpretation by a human observer.

c) Nevn hvilke datatyper som utveksles mellom prosessene i diagrammet som utgjør svaret på deloppgave b), punkt 1. (30 poeng)

Output of image capture -input to preprocessing: grey level/ colour image: arrays of full range pixel values

Output of preprocessing -Input to segmentation: grey level/ colour image arrays of full range pixel values

Output of segmentation -Input to description: label images: arrays of label values, typically small integers well below full range of pixels

Output of description -Input to recognition: lists of labels: a structure, eg. A list, containing numerical or structural features

Output of recognition / measurement: Identity of the object.

OPPGAVE 2 Bildebehandling – Segmentering og beskrivelse (100 poeng)

a) Beskriv to metoder for å forbedre kantene i et bilde. (20 poeng)

The two methods could be taken from:

Sobel masks (Or other mask type: Robell's, Prewitt, etc.)

Difference of Gaussian

Laplacian of Gaussian

Canny

Morphological gradient

High frequency emphasis (weak)

b) Forklar forskjellen mellom begrepet "kant" i et 2-D bilde og det samme begrepet i den virkelige verden. (20 poeng)

An edge in a 2D image is a rapid transition in intensity that is compatible in strength and direction with neighbouring transitions so that, overall, a structure can be inferred.

An edge in the 3D world is a rapid transition in surface normal direction or a depth discontinuity measured from the viewpoint.

The edge in an image is caused by the lighting and reflectivity of the surfaces of the 3D world. An edge in the 3D world often causes an edge in the 2D image. A rapid change in the reflectivity of the surface of an object, without a geometrical transition can also cause an edge in an image, for example, a scanner capturing the image of a document.

c) Svar på følgende spørsmål:

1. Definer hva en forbundet region ("connected region") er. (10 poeng)

A connected region is a set of pixels such that, under some definition of adjacency, it is possible to make a journey from any pixel in the region to any other pixel in the region by making steps between adjacent pixels that all belong to the region. In addition, the pixels of the region share some common property.

2. Skriv pseudokode for groing av en region, R , representert som en mengde av koordinatpar, $\langle x, y \rangle$, basert på medlemskapsbetingelsen $P(R)$. (30 poeng)

Input: definition of adjacency; seed point $A \langle x_0, y_0 \rangle$,

Use: function Neighbours giving the set of all neighbours of its input

Set $R = \{ \langle x_0, y_0 \rangle \}$

Iterate until no change in R {

 Set $\text{new_border_set} = \text{Neighbours}(R) - R$;

 For each element, b , of new_border_set {

 If $P(R \cup b)$ then set $R = R \cup b$

 }

}

- d) Hva menes med "strukturerende element" i morfologi? (20 poeng)

The structuring element in morphology defines the shape that the morphological operations will respond to. It is analogous to the kernel in convolution or the argument of the delta function in the Hough transform. Morphological operations do not have numerical parameters as is common in image processing operations. The structuring element is the parameterisation of the morphological method.

OPPGAVE 3 Bildebehandling – Representasjon

(100 poeng)

- a) Regn ut Euler-tallet for figur 1. (20 poeng)



Figur 1

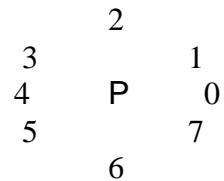
The definition of the Euler number relevant for this question is $E=C-H$ where C is the number of components and H is the number of holes. From the diagram: $C=1$ and $H=2$. Thus $E=-1$.

b) Svar på følgende spørsmål:

1. Definer en 8-veis kjedekode ("chain code"). (20 poeng)

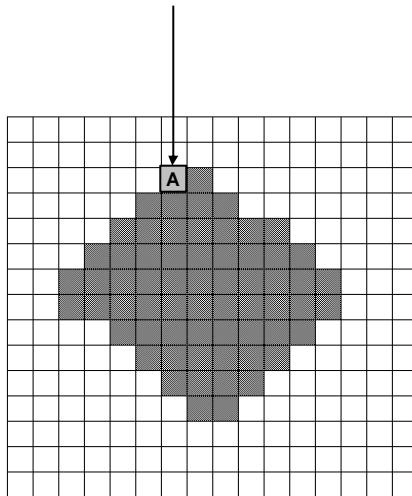
A chain code is a sequence of labels that name a direction from a reference point. The number of different directions is n for an n -way code. Successive steps around a contour generate a sequence of symbols that is the chain code for that contour.

An eight way chain code can be defined by the diagram:



where it is understood that the label is quoted when a step in the direction is made. The label scheme is arbitrary.

2. Finn kjedekoden for konturen av det skraverte området i figur 2. Gå rundt konturen MOT KLOKKA. Start fra piksel **A**. (20 poeng)



Figur 2

Using the chain code above
5555607777011112334334

c) Svar på følgende spørsmål:

1. Definer Fourier-deskriptoren til en form. (20 poeng)

The Fourier descriptor of a shape is obtained by the following steps.

1. List the contour pixels as a sequence of complex numbers, z , where each z is $x+jy$, corresponding to the contour pixels, $\langle x,y \rangle$, taken in order around the shape. Let the length of the list be n .
2. Choose a scale factor so that an angular argument runs $0..2\pi$ in the number of elements of the list.
3. Compute the Fourier transform of the sequence, z , regarding the z as a function of the angular argument.
4. Retain the first m , $m \leq n$, terms of the sequence as a descriptor of the shape. Note that retaining the first m terms preserves the low frequency end of the transform.

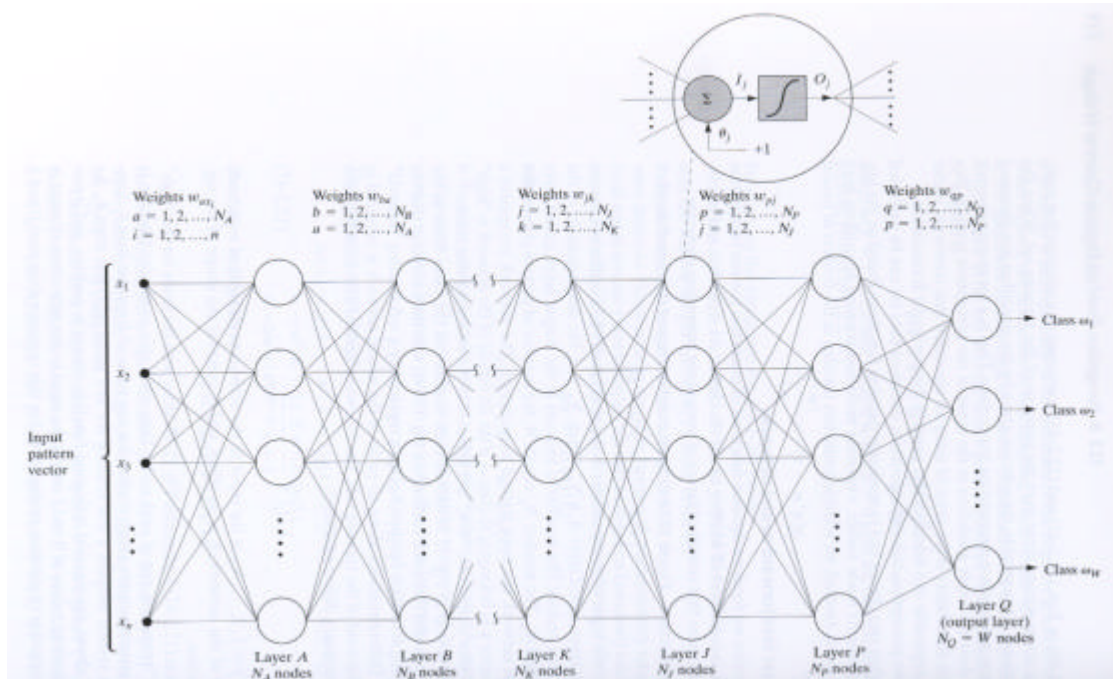
2. Hva skjer med detaljeringsgraden av formen som representeres når stadig flere ledd av Fourier-deskriptoren tas med? (20 poeng)

The low frequency terms are preserved in the Fourier descriptor. As successively more terms are preserved the coverage of the function z in the frequency domain is made more complete. If all the terms of the transform are preserved then, subject to numerical precision, the shape, z , should be recoverable. The preservation of successively more terms allows the finer detail to be recovered. For example, sharp corners are represented with greater precision as more terms are retained.

OPPGAVE 4 Bildebehandling – Representasjon og gjenkjenning (100 poeng)

a) Tegn et diagram som viser strukturen av et nevralt nett. (20 poeng)

An approximation to the book figure 12-16 p 720.



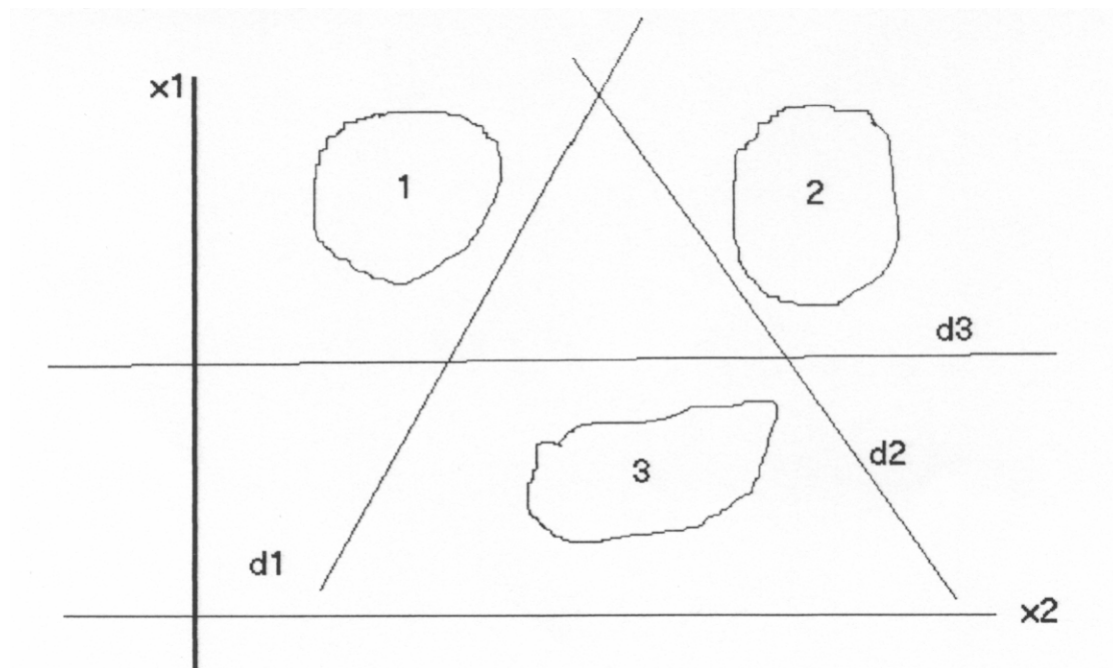
b) Hva menes med å trene et nevralt nett? (20 poeng)

Training a neural net is the process of adjusting the weights so that there is decreasing error between the output values and the target, correct, values. There are several forms of neural net and definitions of the correct functioning. All training methods have repeated presentation of data and a corrective action when an error is detected. The process of convergence is not well understood. It is possible to degrade performance by presenting more correct data; a phenomenon called 'over-training'. This suggests that the theory behind the training methods is incomplete, especially in regard to the final approach to a converged state and the definition of that state.

c) Hvilke begrensninger treffer en på når nevralt nett anvendes på bilder? (20 poeng)

The number of pixels in an image is much larger than the maximum number of input lines to most neural net implementations. For many image processing tasks, the values of most of the pixels are not significant because the domain of interest is a small percentage of the pixel grid. Neural nets applied directly to images face the problem of too large a volume of data and the absence of any criterion, or method, for identifying the area of interest. Without preprocessing to remove these problems training is effectively impossible.

d) Forklar ved hjelp av et diagram hva som menes med en lineær desisjonsflate. (20 poeng)



A linear decision surface is a hyperplane in the feature space (line for 2D, plane for 3D etc.) that partitions the feature space into two parts: one part containing the pattern and the other not. Classification under a linear decision rule is the calculation of the side of the hyperplane on which the unknown point lies. There is no guarantee that a pattern recognition problem is posed in a form such that the classes are linearly separable. Linear separability can usually be obtained by increasing the dimensionality of the feature space but this is often undesirable for practical reasons of increasing the complexity and cost of taking measurements.

e) Hvilken type desisjonsflate blir implementert av et nevralt nett med minst 3 nivå? (20 poeng)

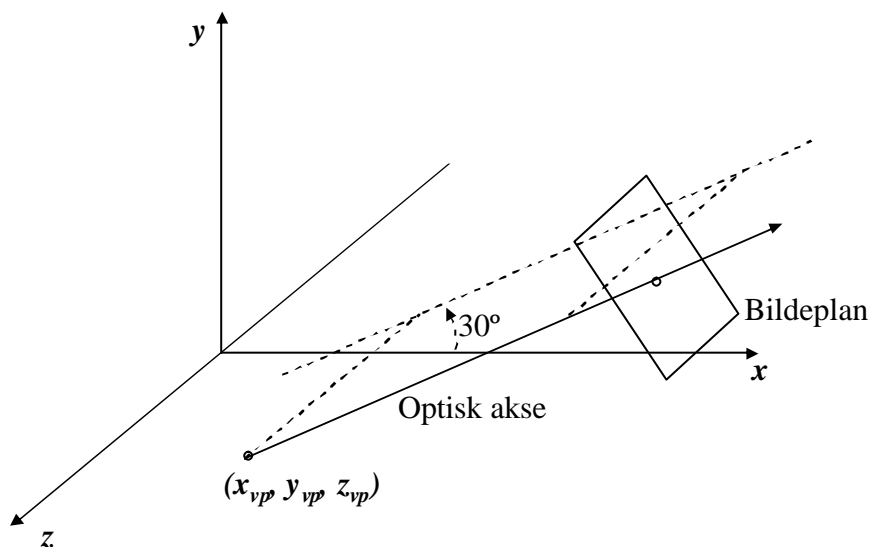
A neural net with at least 3 layers is theoretically able to implement a decision surface that will separate any classes that are not actually degenerate. Obtaining the correct weights by training may be very difficult. In general, it is safe to assume that a piecewise linear decision surface is implemented by a neural net. The extent to which this can approximate an arbitrary decision surface depends on the number of layers, the number of decision units in the layers and the inter-connections.

OPPGAVE 5 Grafikk – Prosjeksjon og kamerakoordinatsystem (100 poeng)

Vi skal foreta en perspektivisk avbildning av en modell med punktet (x_{vp}, y_{vp}, z_{vp}) som projeksjonssenter. Koordinatene er gitt i verdenskoordinatsystemet. Synsretningen (kameraets optiske akse) er parallell med planet $z = 0$ i verdenskoordinatsystemet. Den ortografisk projeksjonen av den optiske akse inn i planet $z = 0$, danner vinkelen 30° med verdenskoordinatsystemets x -akse. Bildeplanet ligger i avstanden d fra projeksjonssenteret målt langs den optiske akse og står vinkelrett på akse.

Det er ingen sammenheng mellom nedenstående deloppgaver a) og b).

- a) Vi ønsker en matrise som gir bildepunktkoordinatene referert til verdenskoordinatsystemet når vi anvender matrisen på objektkoordinater i verdenskoordinatsystemet. Utled matrisen! (50 poeng)



Plan for utledning av avbildningsmatrisen:

1. Transler slik at projeksjonssenteret faller i origo
2. Roter om z -aksen slik at den optiske akse faller langs x -aksen
3. Foreta avbildningen
4. Invers transformasjon av punkt 2.
5. Invers transformasjon av punkt 1.

Hver enkelt skritt:

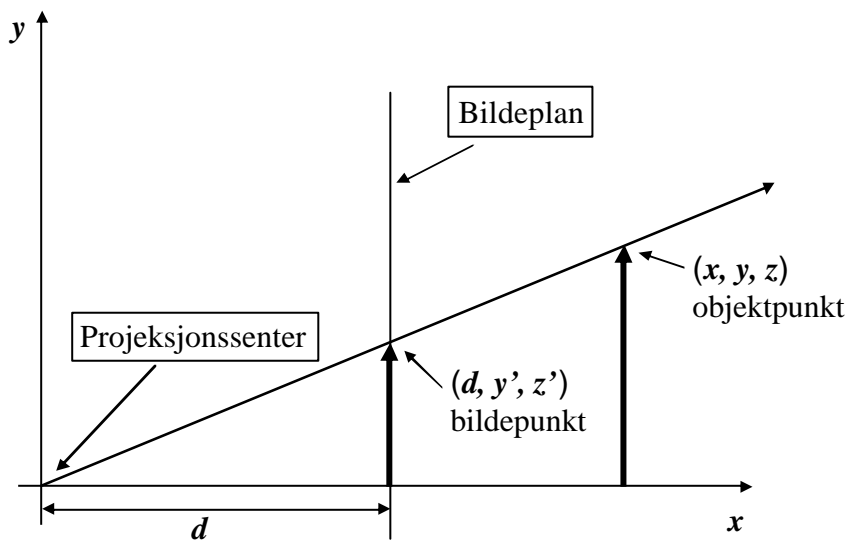
1. Translasjon slik at projeksjonscenteret faller i origo:

$$M_1 = T(-x_{vp}, -y_{vp}, -z_{vp}) = \begin{bmatrix} 1 & 0 & 0 & -x_{vp} \\ 0 & 1 & 0 & -y_{vp} \\ 0 & 0 & 1 & -z_{vp} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2. Rotasjon slik at den optiske akse faller langs x -aksen:

$$M_2 = R_z(-30^\circ) = \begin{bmatrix} \frac{1}{2}\sqrt{3} & \frac{1}{2} & 0 & 0 \\ -\frac{1}{2} & \frac{1}{2}\sqrt{3} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3. Avbildning:



Likedannede trekantar gir:

$$\frac{y'}{d} = \frac{y}{x}$$

$$\Rightarrow y' = \frac{y}{x/d}$$

Tilsvarende:

$$z' = \frac{z}{x/d}$$

Dessuten:

$$x' = d$$

Dette gir matrisen:

$$M_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1/d & 0 & 0 & 0 \end{bmatrix}$$

4. Tilbakerotasjon av den optiske aksens:

$$M_4 = R_z(30^\circ) = \begin{bmatrix} \frac{1}{2}\sqrt{3} & -\frac{1}{2} & 0 & 0 \\ \frac{1}{2} & \frac{1}{2}\sqrt{3} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

5. Tilbaketranslasjon av projeksjonssenteret:

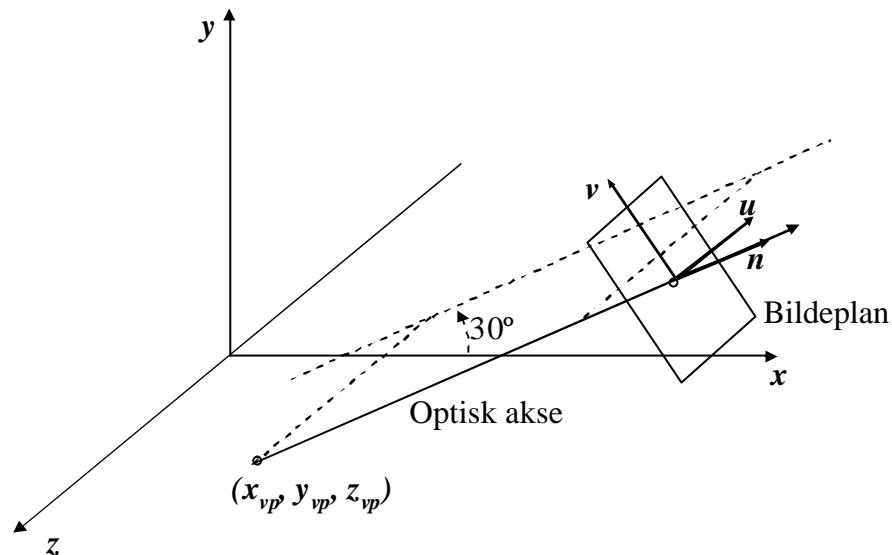
$$M_5 = T(x_{vp}, y_{vp}, z_{vp}) = \begin{bmatrix} 1 & 0 & 0 & x_{vp} \\ 0 & 1 & 0 & y_{vp} \\ 0 & 0 & 1 & z_{vp} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Den søkte avbildningsmatrisen er:

$$M = M_5 \cdot M_4 \cdot M_3 \cdot M_2 \cdot M_1 = \begin{bmatrix} 1 + \frac{x_{vp}}{2d}\sqrt{3} & \frac{x_{vp}}{2d} & 0 & x_{vp}\left(\frac{1}{2} - \frac{x_{vp}}{2d}\sqrt{3} - \frac{y_{vp}}{2d}\right) \\ \frac{y_{vp}}{2d}\sqrt{3} & 1 + \frac{y_{vp}}{2d} & 0 & y_{vp}\left(1 - \frac{x_{vp}}{2d}\sqrt{3} - \frac{y_{vp}}{2d}\right) \\ \frac{z_{vp}}{2d}\sqrt{3} & \frac{z_{vp}}{2d} & 1 & z_{vp}\left(-1 - \frac{x_{vp}}{2d}\sqrt{3} - \frac{y_{vp}}{2d}\right) \\ \frac{1}{2d}\sqrt{3} & \frac{1}{2d} & 0 & -\frac{x_{vp}}{2d}\sqrt{3} - \frac{y_{vp}}{2d} \end{bmatrix}$$

Konkatenering av matrisene forlanges ikke. Ved forsøk på konkatenering sees regnefeil bort fra.

- b) Utled matrisen for transformasjon av verdenskoordinater til kamerakoordinater for den beskrevne konfigurasjonen! (50 poeng)



Kamerakoordinatsystemets akser betegnes som u -, v - og n -aksene. Den søkte matrisen representerer transformasjonen som skal til for å bringe kamerakoordinatsystemets akser til å falle sammen med verdenskoordinatsystemets akser. Kamerakoordinatsystemets n -akse velges langs den optiske akse. Oppgaveteksten gir ingen anvisning for bestemmelse av u - og v -aksene. Disses orientering må derfor velges. Det enkleste valget er å la u -aksen være parallell med verdenskoordinatsystemets x - z -plan. Dermed vil den søkte matrisen komme fram ved stegene 1 og 2 fra deloppgave a) etterfulgt av en rotasjon -90° om y -aksen:

$$M_{xyz \rightarrow uvn} = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} M_2 \cdot M_1 = \frac{1}{2} \begin{bmatrix} 0 & 0 & -2 & 2z_{vp} \\ -1 & \sqrt{3} & 0 & x_{vp} - y_{vp}\sqrt{3} \\ \sqrt{3} & 1 & 0 & -x_{vp}\sqrt{3} - y_{vp} \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

Her bør matrisene kreves korrekt konkatenerert.

OPPGAVE 6**Grafikk – Klipping****(140 poeng)**

a) Liang-Barskys algoritme tar utgangspunkt i linjelikningen på parametrisk form.

1. Hvilken type klippevindu kan algoritmen brukes mot? (20 poeng)

Algoritmen kan brukes mot rektangulære klippevinduer i 2D. (Ikke krevd: i 3D kan den brukes mot rektangulære parallellepiped.)

2. Hvordan konstateres parallellitet med klippevinduets kanter? (20 poeng)

Vi forutsetter at klippevinduet er akseorientert. Da kan parallellitet undersøkes ved sammenlikning av kandidatlinjens endepunktskoordinater. Dersom x -koordinatene er like, er kandidatlinjen parallell med vinduets vertikale kanter. Dersom y -koordinatene er like, er kandidatlinjen parallell med vinduets horisontale kanter.

3. Forklar med støtte i skisser prinsippet for algoritmen! (30 poeng)

Grunnlaget for algoritmen er at den matematiske beskrivelsen av kandidatlinjene er på parametrisk form:

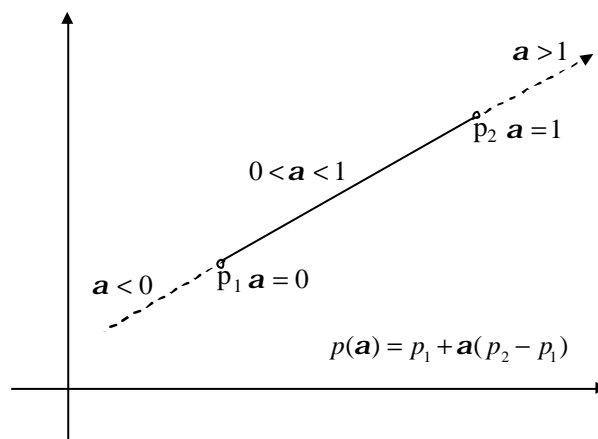
$$p(\alpha) = (1 - \alpha)p_1 + \alpha p_2 = p_1 + \alpha(p_2 - p_1)$$

På komponentform:

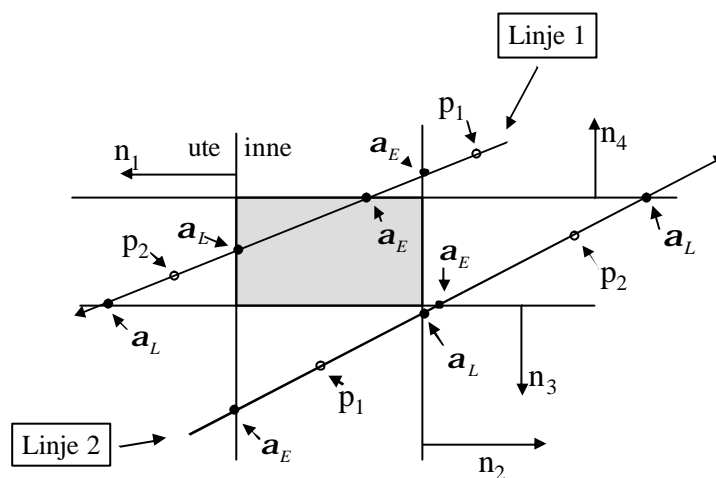
$$x(\alpha) = x_1 + \alpha(x_2 - x_1)$$

$$y(\alpha) = y_1 + \alpha(y_2 - y_1)$$

Tolkningen av parameteren a gir grunnlag for å forestille seg at kandidatlinjen har orientering i retning av stigende a -verdi. a -verdier i intervallet $0 = a = 1$ representerer det linjestykket som er klippekandidat, mens verdier $a < 0$ og $a > 1$ representerer forlengelser av det aktuelle linjestykket.



For hver av klippevinduetts kanter med forlengelser defineres en normal som peker ut fra vinduet. Den andre siden defineres som innsiden. Med mindre de er parallelle med en av klippevinduetts kanter, vil alle linjene som er kandidater for klipping eller deres forlengelser, skjære alle klippevinduetts fire kanter eller deres forlengelser. Parameterverdiene som svarer til de fire skjæringspunktene kan beregnes ved innsetting av henholdsvis de to x -koordinatene og de to y -koordinatene for vinduetts kanter i den parametriske linjelikningen. Når en tar linjens retning i betraktning, vil to av skjæringene representere innpasseringer, a_E , i forhold til kantenes innside og utside, og to av skjæringene vil representere utpasseringer, a_L :



For hver kandidatlinje beholdes den største a_E og den minste a_L . Linjen forkastes i sin helhet i følgende tilfelle:

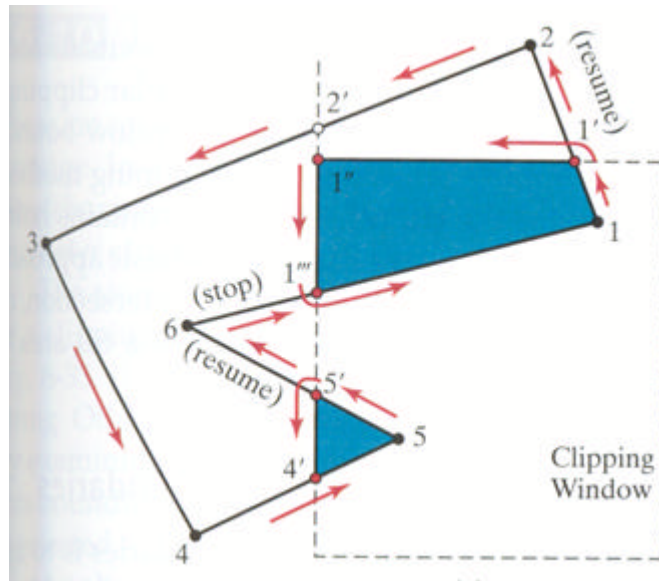
- Linjen er parallell med et par av vinduskantene men ligger helt utenfor vinduet
- $a_E > 1$ (forlengelsen av kandidatlinjen kan skjære vinduet)
- $a_L < 0$ (forlengelsen av kandidatlinjen kan skjære vinduet)
- $a_E > a_L$ (kandidatlinjen ligger helt utenfor vinduet)

For en godtatt linje velges for gjennomføringen av klippingen:

- $a_E = \max(0, a_E)$
- $a_L = \min(1, a_L)$

I implementasjonen av Liang-Barskys algoritme benyttes 8 eller 12 hjelpestørrelser i form av differenser mellom koordinater. Disse betraktes ikke som en del av metoden, men besvarelser som benytter disse størrelsene til å beskrive metoden, kan være fullgode.

- b) Forklar Weiler-Athertons algoritme for polygonklipping ved hjelp av skisse(r) og pseudokode! (70 poeng)



1. Gå rundt polygonet i en bestemt retning (velger her mot klokka) inntil en kant som krysser vinduskanten påtreffes på vei ut
2. Følg kanten til klippevinduet inntil en ny kant av polygonet som klippes, påtreffes. Dersom dette kryssingspunktet er et punkt som er truffet før, gå til neste steg. Dersom kryssingspunktet er et nytt ett, gå videre inntil et hjørne som er behandlet før, påtreffes
3. Bygg opp hjørnelisten for den aktuelle delen av det klipte polygonet
4. Gå tilbake til kryssingspunktet der polygonkanten først krysset ut og følg polygonet videre til hele polygonet er traversert

OPPGAVE 7 Grafikk – Midtpunktsmetoden

(160 poeng)

Bruk midtpunktsmetoden til å utvikle en algoritme for tegning av parabelen:

$$y = x^2$$

i x -intervallet der tangenten til parabelen har stigningsforhold mellom 0 og 1. (160 poeng)

Grunnlaget for bruk av midtpunktsmetoden er kurvelikningen på implisitt form. Den gitte likningen omskrevet til implisitt form blir:

(1) $f(x, y) = x^2 - y = 0$

For å gjøre løsningen mer generell, kan vi generalisere parabellikningen til:

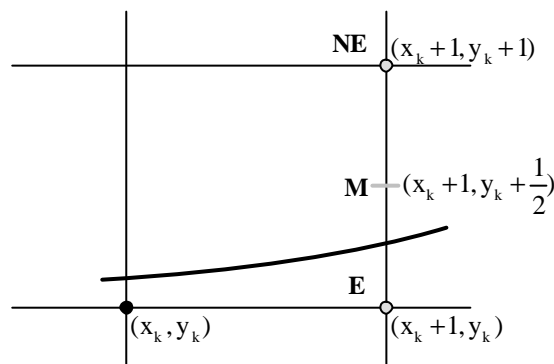
$$(2) \quad f(x, y) = px^2 - y = 0$$

Denne generaliseringen er ikke et krav i oppgaven, og besvarelser som ikke gjør dette, kan likevel være fullgode.

Koeffisienten p i likning (2) er generelt et flyttall. Imidlertid er flyttall som kan representeres i en datamaskin, alltid rasjonale tall. Derfor kan vi sørge for at likningen erstattes med en likeverdig likning med heltallskoeffisienter ved å skalere med et passende positivt heltall $2b$ slik at vi får:

$$(3) \quad f(x, y) = ax^2 - 2by = 0$$

der koeffisientene a og b er heltall. Faktoren 2 er med for å sikre at beregningene i det følgende kan utføres som heltallsberegninger. Et viktig poeng med midtpunktmetoden er at alle operasjoner kan utføres som heltallsoperasjoner.



Om vi for en gitt x -verdi setter inn i likning (3) y -verdien for et punkt som ligger over kurven, ser vi av likningen at vi får $f(x, y) < 0$. Setter vi inn y -verdien for et punkt som ligger under kurven, vil vi få $f(x, y) > 0$. Dette kan vi utnytte. Anta at det siste pikselet som er valgt, er (x_k, y_k) . Med stigningsforhold som forutsatt mellom 0 og 1 vil da pikslene som er kandidater for å bli valgt på neste nettligne $x_{k+1} = x_k + 1$ være pikslene $E = (x_k + 1, y_k)$ og

$NE = (x_k + 1, y_k + 1)$. Vi ser på midtpunktet $M = (x_k + 1, y_k + \frac{1}{2})$ mellom de to

kandidatpikslene. Dersom det ligger over den matematisk nøyaktige kurven, velges pikselet E. Dersom det ligger under, velges pikselet NE. Beliggenheten av midtpunktet M i forhold til den matematisk nøyaktige kurven kan undersøkes ved hjelp av en desisjonsvariabel, d_k :

$$(4) \quad d_k = f(M)$$

Vi får:

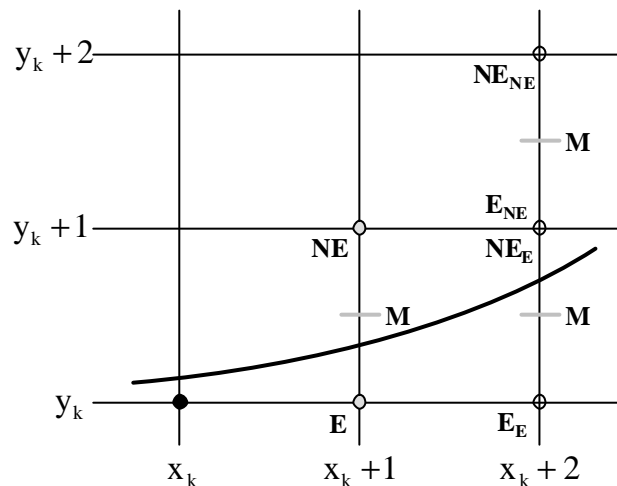
$$(5) \quad d_k = a(x_k + 1)^2 - 2b(y_k + \frac{1}{2})$$

$$(6) \quad d_k \leq 0 \quad \Rightarrow \quad E = (x_k + 1, y_k) \text{ velges}$$

$$(7) \quad d_k > 0 \quad \Rightarrow \quad NE = (x_k + 1, y_k + 1) \text{ velges}$$

At nedre piksel velges når $d_k = 0$, er i dette tilfelle et vilkårlig valg.

Beregningen av d_k etter likning (5) krever tre multiplikasjoner og tre addisjoner eller subtraksjoner. Beregningen kan forenkles og effektiviseres ved å se på hva som skjer på neste nettlinj $x_{k+2} = x_k + 2$:



Valget på nettlinj $x_{k+2} = x_k + 2$ er avhengig av hvilket valg som blir gjort på nettlinj $x_{k+1} = x_k + 1$. Dersom E- pikselet blir valgt, er de nye kandidatpikslene $E_E = (x_k + 2, y_k)$ og $NE_E = (x_k + 2, y_k + 1)$. Dersom NE-pikselet blir valgt, er de nye kandidatpikslene $E_{NE} = (x_k + 2, y_k + 1)$ og $NE_{NE} = (x_k + 2, y_k + 2)$. Den nye verdien til desisjonsvariabelen som beregnes på etter valget på nettlinj $x_{k+1} = x_k + 1$ blir dermed:

Dersom E-pikselet blir valgt:

$$(8) \quad \begin{aligned} \underline{d_{k+1}} &= a(x_k + 2)^2 - 2b(y_k + \frac{1}{2}) = a(x_k + 1)^2 - 2b(y_k + \frac{1}{2}) + 2x_k + 3 = \\ &= \underline{d_k + 2ax_k + 3} = d_k + \Delta_{E,k} \end{aligned}$$

med:

$$(9) \quad \Delta_{E,k} \equiv 2ax_k + 3$$

Dersom NE-pikselet blir valgt:

$$(10) \quad \begin{aligned} \underline{d_{k+1}} &= a(x_k + 2)^2 - 2b(y_k + \frac{3}{2}) = a(x_k + 1)^2 - 2b(y_k + \frac{1}{2}) + 2x_k + 3 - 2b = \\ &= \underline{d_k + 2ax_k + 3 - 2b} = d_k + \Delta_{NE,k} \end{aligned}$$

med:

$$(11) \quad \Delta_{NE,k} \equiv 2ax_k + 3 - 2b$$

I oppgaven er $a = 0$ og $b = \frac{1}{2}$. Med disse verdiene blir de to uttrykkene for de nye desisjonsvariablene:

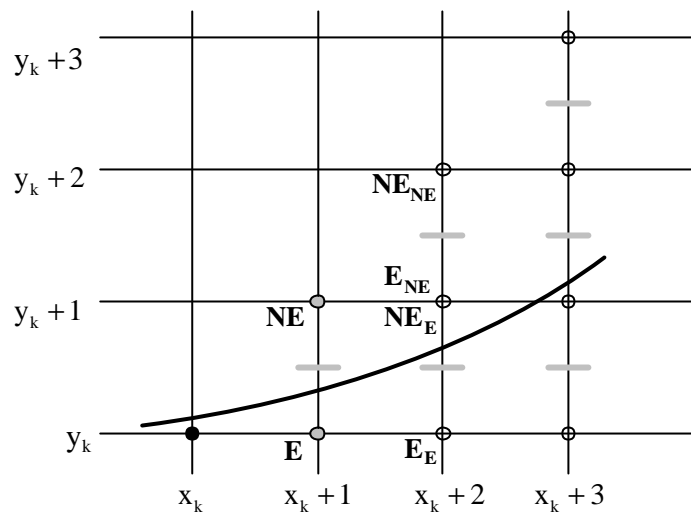
$$\begin{aligned} d_{k+1} &= d_k + 2x_k + 3 && \text{dersom E-pikselet blir valgt} \\ d_{k+1} &= d_k + 2x_k + 2 && \text{dersom NE-pikselet blir valgt} \end{aligned}$$

Dette er fullgode resultater så langt.

Den nye verdien av desisjonsvariablen kan altså beregnes ved inkrementering av den gamle. Omfanget av beregninger er redusert til en multiplikasjon og to addisjoner.

Besvarelser som når så langt som hit og som i tillegg bestemmer startverdi for desisjonsvariablen og setter opp algoritmen (se til siste del av løsningsforslaget), bør gies 120 av de 160 poengene. For å få full uttelling må også beregning av $\Delta_{E,k}$ og $\Delta_{NE,k}$ ved inkrementering være med.

På nettlinsen x_k beregnes verdien av desisjonsvariablen d_{k+1} som skal brukes på nettlinsen $x_{k+1} = x_k + 1$ til å velge piksel på nettlinsen $x_{k+2} = x_k + 2$. Vi slo fast i likningene (8) og (10) at denne verdien er avhengig av hvilket piksel, N eller NE, som velges på nettlinsen $x_{k+1} = x_k + 1$. På samme måte beregnes på nettlinsen $x_{k+1} = x_k + 1$ verdien av desisjonsvariablen d_{k+2} som skal brukes på nettlinsen $x_{k+2} = x_k + 2$ til å velge piksel på nettlinsen $x_{k+3} = x_k + 3$.



Vi ser på inkrementene til desisjonsverdiene:

Dersom piksel E ble valgt på nettlinje $x_{k+1} = x_k + 1$ får vi to tilfelle:

Dersom piksel E_E skulle bli valgt på nettlinje $x_{k+2} = x_k + 2$:

$$(12) \quad \underline{\Delta_{E,k+1}} = 2ax_{k+1} + 3 = 2a(x_k + 1) + 3 = 2ax_k + 3 + 2a = \underline{\Delta_{E,k} + 2a}$$

Dersom piksel NE_E skulle bli valgt på nettlinje $x_{k+2} = x_k + 2$:

$$(13) \quad \underline{\Delta_{NE,k+1}} = 2ax_{k+1} + 3 - 2b = 2a(x_k + 1) + 3 - 2b = 2ax_k + 3 - 2b + 2a = \underline{\Delta_{NE,k} + 2a}$$

Dersom piksel NE ble valgt på nettlinje $x_{k+1} = x_k + 1$ får vi også to tilfelle:

Dersom piksel E_{NE} skulle bli valgt på nettlinje $x_{k+2} = x_k + 2$:

$$(14) \quad \underline{\Delta_{E,k+1}} = 2ax_{k+1} + 3 = 2a(x_k + 1) + 3 = 2ax_k + 3 + 2a = \underline{\Delta_{E,k} + 2a}$$

Dersom piksel NE_{NE} skulle bli valgt på nettlinje $x_{k+2} = x_k + 2$:

$$(15) \quad \underline{\Delta_{NE,k+1}} = 2ax_{k+1} + 3 - 2b = 2a(x_k + 1) + 3 - 2b = 2ax_k + 3 - 2b + 2a = \underline{\Delta_{NE,k} + 2a}$$

For vår parabel ser vi at vi alltid får nytt inkrement ved å øke det gamle med $2a$, eller 2 dersom vi strengt holder oss til oppgaveformuleringen.

Vi trenger startverdi for desisjonsvariabel og inkrement. Vi antar at vi starter tegning av parablen i origo $(0, 0)$. Vi får da startverdiene:

$$(16) \quad \underline{d_0 = a - b}$$

$$(17) \quad \underline{\Delta_{E,0} = 3}$$

$$(18) \quad \underline{\Delta_{NE,0} = 3 - 2b}$$

Algoritme for parabeltegning i pseudokode basert på ovenstående analyse og på at den foreliggende parablen er symmetrisk om y-aksen:

1. Sett $x_0 = y_0 = 0$
2. Initier d_0 , $\Delta_{E,0}$ og $\Delta_{NE,0}$ i samsvar med likningene (16) – (18)
3. Sett $k = 0$
4. Repeter:
 - i. Plott piksel (x_k, y_k) og dersom $k > 0$ piksel $(-x_k, y_k)$
 - ii. Sett $x_{k+1} = x_k + 1$
 - iii. Avbryt når stigningsforholdet overskriver 1
 - iv. Bruk d_k til å bestemme y_{k+1} i samsvar med likningene (6) og (7)
 - v. Bruk likning (8) eller (10) avhengig av valget av y_{k+1} til å bestemme d_{k+1}
 - vi. Bruk likningene (12) og (13) eller (14) og (15) avhengig av valget av y_{k+1} til å bestemme $\Delta_{E,k+1}$ og $\Delta_{NE,k+1}$
 - vii. Sett $k = k + 1$

Merknad:

Kandidater som bemerker at for den gitt parablen overstiger stigningsforholdet 1 allerede for $x = 1$, bør få plusspoeng for dette.