



Institutt for datateknikk
og informasjonsvitenskap

**EKSAMEN I EMNE TDT4195
BILDETEKNIKK
ONSDAG 3. JUNI 2009
KL. 09.00 – 13.00**

LØSNINGSFORSLAG

OPPGAVE 1 Grafikk – diverse spørsmål

- a) Primitiver er de grafiske entitetene som skal tegnes. Attributer angir hvordan primitivene skal tegnes/se ut.
Eksempler på primitiver : punkter, linjer, kurver, flater, bokstaver, tegn, osv.
Eksempler på attributer : farge, linjetykkelse, linjetype, flatemønster, font, osv.

Metoder for innlemming av attributter:

Attributer som parametere for hver primitiv.

Ha en liste over gjeldene attributer, tilstandsliste – state (som i opengl)

- b) Konveks polygon: Alle indre vinkler mellom to påfølgende kanter er mindre enn 180 grader.
Konkav polygon: En eller flere indre vinkler mellom to påfølgende kanter er større enn eller lik 180 grader

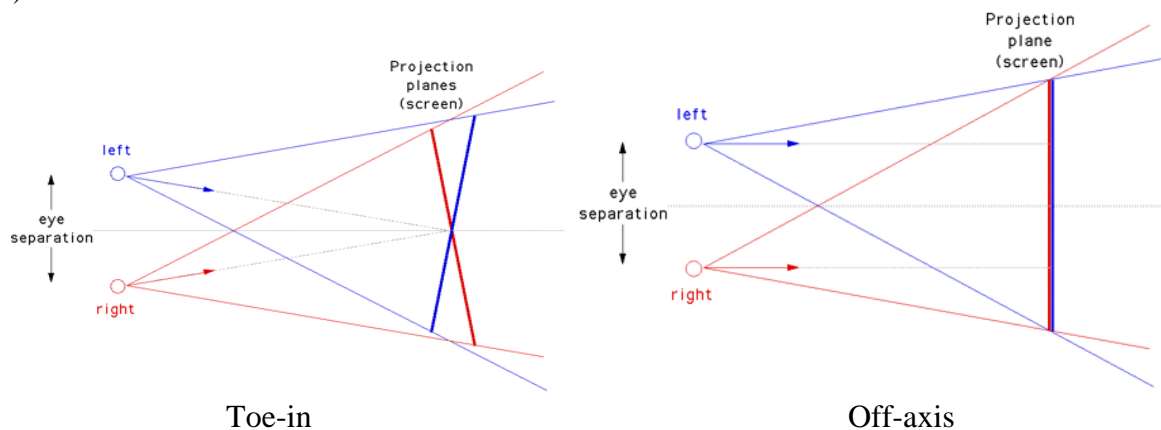
Metode 1: Sett opp en vektor for hver kant, ta vektorproduktet mellom to påfølgende kanter. Dersom alle vektorprodukt har samme fortegn, så er polygonen konveks. Dersom vektorproduktene har forskjellige fortegn, er polygonen konkav.

Metode 2: Utvid hver kant og sjekk for hver kant at alle hjørnene i polygonen ligger på samme side av den utvidede kanten. Da er polygonen konveks. Dersom det finnes hjørner på begge sider av en utvidet kant, er polygonen konkav.

- c) Vi antar at polygonkantene er orientert slik at marsjretningen rundt polygonen er mot klokken (CCW). Vi kan da finne normalen til polygonen ved hjelp av et kryssprodukt som for eksempel $N = (P_2 - P_1) \times (P_3 - P_1)$.

I kamerarommet er kameraet posisjonert i origo (projeksjonssenter i origo eller bildeplan gjennom origo). Vi velger et vilkårlig punkt på polygonen som for eksempel P_1 . En vektor fra kameraet mot polygonen er da $V = (P_1 - 0)$. Dersom vinkelen mellom N og V er mindre enn 90 grader eller større enn 270 grader, vender polygonen vekk fra kameraet. I disse tilfellene er skalarproduktet mellom N og V er positivt. Metoden kan brukes til å forkaste polygoner som ikke kan ses av kameraet, (back-face culling).

d)



Ved "toe-in" benyttes symmetriske projeksjonsfrustra. Symmetriaksene for de to kameraene er normaler til projeksjonsplanet. De dreies på en symmetrisk måte mot henholdsvis venstre og høyre slik at de skjærer hverandre på projeksjonsskjermen. Dette medfører at projeksjonsplanene ikke faller sammen. Dette introduserer feil ut mot kantene.

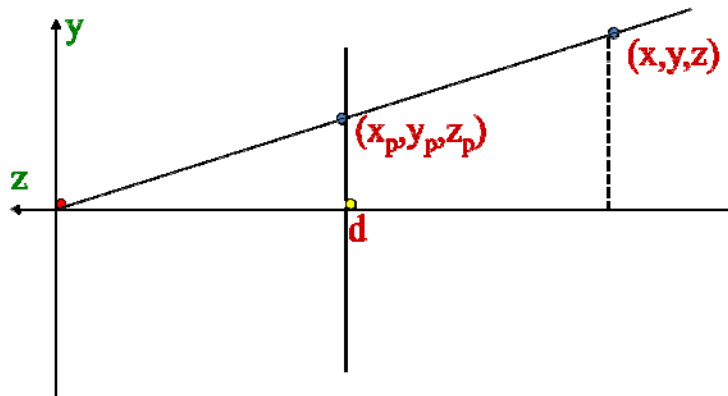
Ved "off-axis" nyttes skjeve frustra for å få sammenfallende projeksjons-plan. Aksene er parallelle og er rettet loddrett mot projeksjonsskjermen. Slik unngår en feil ut mot bildekanten.

e) Mulige grunner:

- Plassbesparende og mer effektivt i forhold til matriser (men matriseform må brukes ved konkatenering)
- Glatt interpolasjon av rotasjon langs en bane er lett å gjennomføre
- Unngår kardanglåsing (gimbal lock)

OPPGAVE 2 Grafikk – projeksjon, refleksjonsmodell og rastring

a)



Bildet av punktet (x, y, z) er (x_p, y_p, z_p) . Av figuren ser vi:

$$\frac{y_p}{y} = \frac{d}{z} \quad \Rightarrow \quad y_p = \frac{y}{z/d}$$

Tilsvarende:

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

Dessuten har vi:

$$z_p = d$$

Vi arbeider med homogene koordinater. Setter vi:

$$w = z/d$$

får vi:

$$x_p = \frac{x}{w}, \quad y_p = \frac{y}{w}, \quad z_p = \frac{z}{w}$$

På matriseform:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ z/d \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

Den søkte matrisen er:

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

b) Phongs refleksjonsmodell:

$$I_{\lambda} = \sum_i \left\{ \frac{S_i}{a + bd_i + cd_i^2} \left[k_{\lambda d} I_{i\lambda d} (l_i \cdot n) + k_{\lambda s} I_{i\lambda s} (r_i \cdot v)^{\alpha} \right] + k_{\lambda a} I_{i\lambda a} \right\} + k_{\lambda a} I_{\lambda a}$$

Leddene som inngår:

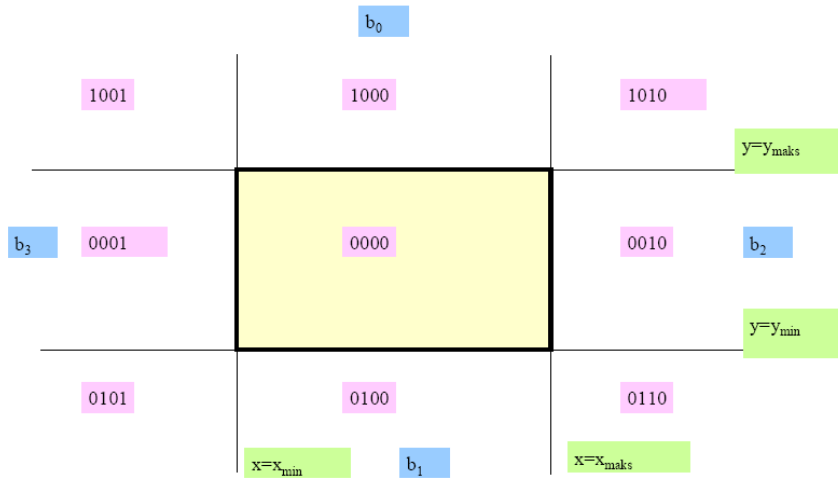
- Leddet $k_{\lambda d} I_{i\lambda d} (l_i \cdot n)$ modellerer diffus spredning.
- Leddet $k_{\lambda s} I_{i\lambda s} (r_i \cdot v)^{\alpha}$ modellerer skinn (refleksjon) fra en flate som har en grad av glans.
- Leddet $k_{\lambda a} I_{i\lambda a}$ er bidraget til bakgrunnsbelysningen forårsaket av lyskilde i .
- Leddet $k_{\lambda a} I_{\lambda a}$ er den globale bakgrunnsbelysningen.
- Faktoren $\frac{1}{a + bd_i + cd_i^2}$ modellerer avstanden til og utstrekningen av lyskilde i . d er

avstanden til lyskilden. For en punktluskilde er $b = 0$. Faktoren kan tilpasses en lyskilde med utstrekning ved å velge størrelsen av koeffisienten b . Hensikten med koeffisienten a er å hindre at faktoren blir for stor ved små avstander d .

Symbolene for øvrig har følgende betydning:

| | |
|------------|--|
| I | - lysstyrke |
| i | - summerer over alle lyskildene |
| λ | - summerer for hver fargekomponent r , g og b |
| $I_{i..}$ | - belysningskomponent fra lyskilde i |
| S_i | - okklusjonsfaktor for lyskilde i (=1 synlig, =0 usynlig) |
| k | - refleksjonskoeffisient |
| α | - "glanstall", stor verdi gir smal refleksjonslobe (opp mot speilende refleksjon), liten verdi gir bred refleksjonslobe (stor spredning) |
| indeks d | - diffus refleksjon |
| indeks s | - speilende refleksjon |
| indeks a | - bakgrunnsbelysning |
| n | - flatenormal, enhetsvektor |
| l_i | - enhetsvektor i retning lyskilde i |
| r_i | - enhetsvektor i retning av perfekt reflektert stråle fra lyskilde i |
| v | - enhetsvektor i retning mot projeksjonssenteret (øyet) |

- c) Cohen-Sutherlands algoritme for linjeklipping i 2D benytter en deling av planet i 9 regioner ved at klippevinduet kanter forlenges, se figur 1.

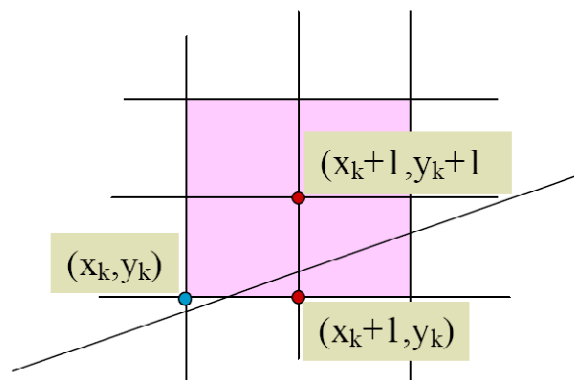


Figur 1

Hver sone tildeles en firebits ”utkastingskode”. Koden bygges opp ved at sonene tildeles en bitposisjon basert på hvilken side den ligger i forhold til en kant av klippevinduet. For eksempel kan sonene over klippevinduet tilordnes bit 0, sonene under bit 1, sonene til høyre bit 2 og sonene til venstre bit 3. Satt bit lik 1 i gitt posisjon viser beliggenhet i forhold til vedkommende kant. Utkastingskoden for selve vinduet har alle fire bit satt lik 0. Se eksempelkode i figuren over.

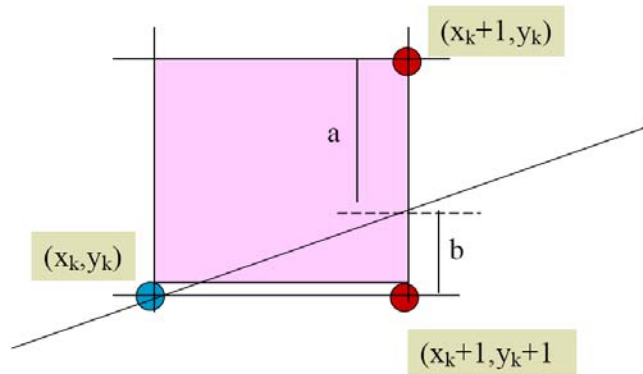
I 3D kan en dele rommet inn i regioner rundt det normaliserte betraktningvolumet (klippevolumet). Dette gjøres ved å utvide planene som utgjør volumets sider i stedet for linjene som utgjør klippevinduet i 2D. Man trenger da 6 bits i stedet for 4 bit for utkastingskoden og klipping vil skje mellom linjen og planene.

- d) Vi begrenser oss til å betrakte linjer med stigningsforhold $m \in [0, 1]$. Linjer med andre stigningsforhold kan skankonverteres på tilsvarende måte.



Vi har som oppgave å skankonvertere den rett linjen $y = mx + h$ fra punktet (x_1, y_1) til punktet (x_2, y_2) . Pikselen (x_k, y_k) er den siste pikselen som ble ”slått på”. Det er to

kandidatpikslar for det neste å "slå på": $(x_k + 1, y_k)$ og $(x_k + 1, y_k + 1)$. Vi definerer desisjonsvariabelen: $d = a - b$:



Vi velger den pikselen som ligger nærmest den matematiske linjen:

$d > 0$: nedre piksel velges
 $d \leq 0$: øvre piksel velges

Den nøyaktige y -verdien på "nettlinsen" $x_k + 1$ får vi ved innsetting i linjelikningen:

$$y = m(x_k + 1) + h$$

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)}$$

a og b kan uttrykkes:

$$a = (y_k + 1) - y = y_k + 1 - m(x_k + 1) - h$$

$$b = y - y_k = m(x_k + 1) + h - y_k$$

Dette gir for d :

$$d = a - b = 2y_k - 2m(x_k + 1) - 2h + 1$$

Vi ordner oss slik med linjens endepunkter at $x_2 > x_1$. Vi kan da danne oss en ny desisjonsvariabel d' ved å multiplisere d med faktoren $x_2 - x_1$:

$$\begin{aligned} d' &= (x_2 - x_1)d = 2y_k(x_2 - x_1) - 2x_k(y_2 - y_1) - 2(y_2 - y_1) - (2h - 1)(x_2 - x_1) = \\ &= 2y_k(x_2 - x_1) - 2x_k(y_2 - y_1) + c \end{aligned}$$

$$c = -2(y_2 - y_1) - (2h - 1)(x_2 - x_1)$$

Siden både x_1, x_2, x_k, y_1, y_2 og y_k forutsetningsvis er heltall, er også de to første leddene i uttrykket for d' heltall. Siden h inngår i konstanten c og h ikke nødvendigvis er heltall, er c i alminnelighet heller ikke heltall. I den videre utviklingen av algoritmen elimineres også dette problemet. Men det er ikke en del av denne oppgaven

OPPGAVE 3 Bildebehandling – Fourier transformen og frekvensdomenefiltrering

a) Proof:

Consider $F(u + N, v)$. Substituting $u + N$ for u in the definition gives:

$$F(u + N, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \exp(-2\pi j(u + N)x / N) \exp(-2\pi jvy / N) f(x, y)$$

Expanding the term

$$\exp(-2\pi j(u + N)x / N)$$

gives

$$\exp(-2\pi jux / N) \exp(-2\pi jNx / N)$$

or

$$\exp(-2\pi jux / N) \exp(-2\pi jx)$$

But x is an integer so

$$\exp(-2\pi jx) = 1$$

Thus

$$F(u + N, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \exp(-2\pi jux / N) \exp(-2\pi jvy / N) f(x, y)$$

$$F(u + N, v) = F(u, v) \quad \text{QED.}$$

b)

$$2\pi ux / N = 12\pi x / N$$

$$2\pi vy / N = 16\pi y / N$$

Solving for u and v :

$$u = 6$$

$$v = 8$$

Thus $(\pm 6, \pm 8)$ are peaks.

c) The radius is the hypotenuse of a right angle triangle with sides 6 and 8. The length of the hypotenuse is 10.

d) Filters that do not have a smooth profile generate 'ringing'.

OPPGAVE 4 Bildebehandling – grunnleggende bildebehandling og filtrering

a) A light table is similar to a slide viewer. It is a translucent surface that is illuminated from beneath. Objects placed on the surface appear in silhouette if a camera is placed above them.

The transition in intensity between the back-lit translucent surface and the object is generally easy to detect and the complications of reflected light are avoided.

b) Diagram should show that the distance between a corner of a pixel and the centre of an 8-neighbour is $\sqrt{2}h$.

c) There are several possibilities:

- Set up camera and objects to be observed so that the narrowest width of any structure is $\sqrt{2}h$ when projected on the pixel grid of pixels with side h .
- Set up the camera and objects to be observed so that the sampling interval, $\sqrt{2}h$, defined by the worst case distance between centres on their pixel grid, is smaller than the half-period of the highest spatial frequency present in a Fourier Transform of a finely sampled version of the image.

d) Let $F(u) = T(f(x))$ where T is the transformation.

T is linear if

$$T(af(x) + bg(x)) = aF(u) + bG(u)$$

where a and b are constants.

e) Median filtering replaces the centre pixel of a patch with odd numbers of pixels on the side, eg. 3x3 or 5x5, with the pixel value that is the median of all the pixels in the patch.

The result of this process is that there is a slight blurring of the image.

OPPGAVE 5 Bildebehandling – segmentering, beskrivelse og gjenkjennelse

- a) Region based segmentation gathers adjacent pixels together to form regions based on a uniformity criterion. A region is defined as a set of adjacent pixels that share a common property. The common property is that the uniformity criterion is satisfied. Region based segmentation can operate under several different control regimes: direct extraction, region growing, split and merge.
- direct extraction covers an operation such as thresholding where pixels are classified individually and then connected sets are identified as regions.
 - region growing begins with a small collection of seed pixels. The pixels neighbouring the contour of the embryonic region are added to the region if doing so would not void the uniformity criterion for the region as a whole. The process continues until there can be no more additions.
 - split and merge operations combine smaller neighbouring regions if the composite region satisfies the uniformity criterion. Regions can be split if the re-arrangement allows a better overall fit to the uniformity criterion. Split and Merge is a difficult process to set parameters for.
- b) Thresholding is an application of statistical decision theory. A decision threshold that separates foreground and background can be chosen based on a number of criteria. A simple criterion, that works well when the numbers of foreground and background pixels are approximately equal, is to choose the crossover point ie. the intensity bucket where the number of pixels associated with the foreground and background, in that bucket, are approximately equal.
- c) There are several methods:
- use of edges to mask out the sections of the image where there may be a transition between classes.
 - Otsu's method
 - Kittler/Illingworth method.
- d) $E = C - H =$
- e) A cluster in feature space is a set of examples of a class, plotted in the feature space. The density of the plotted points is effectively the probability density function of the class if a large number of examples is present.