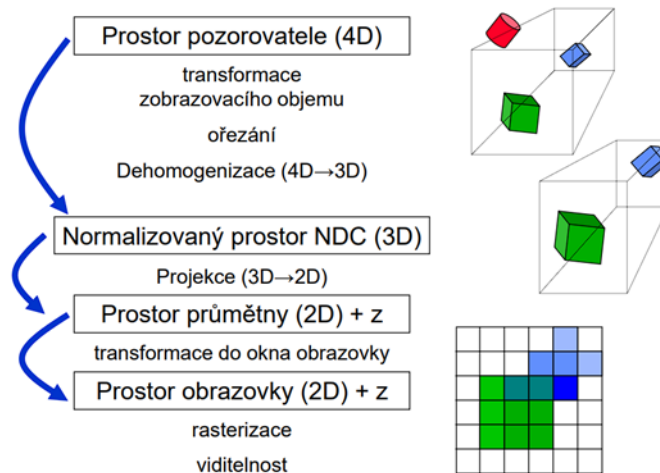


A. Vizualizační řetězec

1. Vizualizační řetězec

- proces, který transformuje 3D objekty do 2D prostoru (obrazovka)
- kombinace několika transformací:
 - modelovací transformace (tvar/pozice objektu)
 - kamerová transformace (projekční)
 - viewport transformace (zobrazovacího objemu)

Zobrazovací řetězec II (pro Zbuffer)



a) Homogenní souřadnice, použití, výhody, ořezávání

- souřadnice skládají se ze 4 hodnot
 - x, y, z, w
- umožňuje transformaci bodů pomocí maticového násobení
- ořezání se provádí na základě hodnoty w
 - body sw menší než z -near nebo větší než z -far jsou považovány za body mimo zobrazovací objem

b) Modelovací transformace, zápis transformací, matice

- proces transformace objektů ve 3D prostoru do virtuálního modelu
 - slouží k umístění objektů do správného místa
 - změně jejich velikosti
 - orientace
 - translace, rotace, škálování
- zapsány pomocí matic
 - čtvercová matice
 - reprezentace a aplikace jedné nebo více transformací
 - 4x4 matice, která kombinuje jednotlivé transformační operace

$$\begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

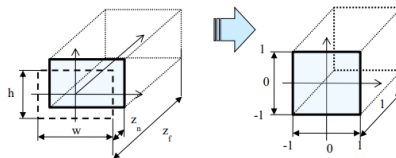
- $R_{11} - R_{33}$ jsou prvky matice rotace
- T_x, T_y, T_z jsou hodnoty translace
- poslední řádek 0, 0, 0, 1 slouží k zachování homogenních souřadnic

c) Pohledová transformace, definice soustavy souřadnic

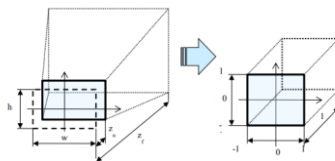
- určuje jak je scéna pozorována ze zvoleného místa a úhlu
 - definuje souřadnicový systém, který popisuje, jak se 3D scéna má zobrazit na 2D rovině
- provádí se po modelovací transformaci, před projekční transformací
- obvykle v pravoúhlé kartézské soustavě
 - 3 osy
 - x – horizontální
 - y – vertikální
 - z – osa, která směřuje do kamery nebo z ní ven
- soustava souřadnic kamery se skládá z
 - polohy v prostoru
 - natočení kamery

d) Transformace zobrazovacího objemu, definice, příklady

- definování oblasti ve 3D prostoru, která bude viditelná na 2D obrazovce
- několik typů projekce
 - ortogonální
 - vytváří plochý pohled na scénu
 - objekty mají konstantní velikost bez ohledu na jejich vzdálenost od kamery
 - tvar kvádru



- perspektivní
 - objekty se zmenšují v závislosti vzdálenosti od kamery
 - používá se k vytvoření dojmu prostoru a hloubky scény
 - tvar komolého jehlanu



- ohraničena šesti parametry
 - x_{min}
 - x_{max}
 - y_{min}
 - y_{max}
 - z_{near}
 - z_{far}
- body mimo zobrazovací objem se ořezávají → nejsou zobrazeny

e) Transformace do okna

- poslední fáze zobrazení 3D scény na 2D obrazovku
- převádí normalizované souřadnice zobrazovacího objemu na souřadnice obrazovky
- transformuje pouze x, y souřadnice
 - souřadnici z uchovááme pro řešení viditelnosti
- upraví se poměr stran, aby se 2D obraz zobrazil v celém okně

f) Algoritmus Z-buffer, implementace

- ukládá hodnotu hloubky (z-hodnotu) každého pixelu
 - hodnotu porovná s aktuální hodnotou v z-bufferu
 - pouze pixely s nižší z-hodnotou jsou vykresleny a aktualizují hodnoty z-bufferu
- implementace
 - transformace – 3D homogenní souřadnice
 - tvar objektu
 - pozice objektu
 - pohledu
 - zobrazovacího objemu
 - ořezání homogenní souřadnice
 - dehomogenizace
 - odstranění souřadnice w
 - vydělení všech souřadnic hodnotou w
 - projekce
 - $3D \rightarrow 2D$
 - transformace do okna
 - rasterizace
 - viditelnosti
 - porovnání hodnoty v z-bufferu

g) Mapování textury, souřadnice do textury, interpolace

- možnost přidání detailu, barvy nebo vzoru na povrch 3D objektů pomocí 2D textur
 - při mapování jsou texturovací souřadnice přiřazeny ke každému bodu nebo vrcholu objektu
 - souřadnice jsou v rozsahu 0 – 1
 - určují, jaký pixel z textury se má použít pro vykreslení daného bodu, nebo vrcholu
- interpolace se provádí pomocí barycentrického zobrazení
 - výpočet váhy pro každý vrchol a použití k interpolaci hodnot v daném pixelu

h) Ořezání zobrazovacím objemem

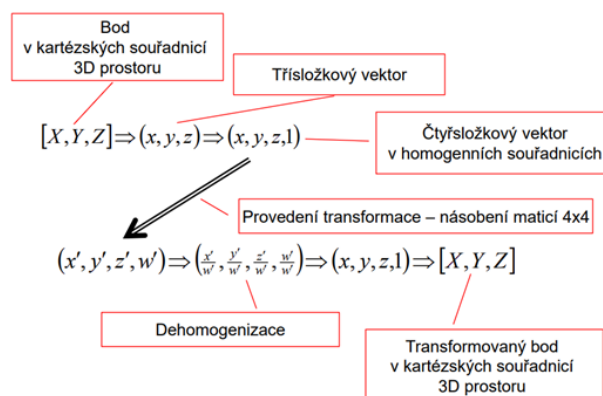
- cílem je nezpracovávat část scény ležící mimo zobrazovací objem
 - zlepšení výkonu
 - efektivnější vykreslení scény
- vrchol x, y, z po dehomogenizaci v NDC ořezán podmínky
 - $-1 < x < 1$
 - $-1 < y < 1$
 - $0 < z < 1$
- vrchol x, y, z, w před dehomogenizací ořezán podmínky
 - $-w < x < w$

- $-w < y < w$
 - $0 < z < w$
- postup ořezání:
 - bod – kontrola podmínek
 - úsečka – kontrola vrcholů a zjištění části úsečky
 - trojúhelník
 - může vzniknout až devítiúhelník
 - Sutherland-Hodgman algoritmus
- způsoby:
 - přísné ořezání
 - jakmile jeden bod leží mimo zobrazovací objem, trojúhelník / úsečku zahazujeme
 - rychlé ořezání
 - zjistíme, zda celý objekt neleží mimo zobrazovací objem, pokud ano zahodíme, pokud ne pokračujeme ve zpracovávání
 - nutno odřezat část ležící za pozorovatelem, záporné nebo nulové w , z důvodu dehomogenizace = dělení w

i) Jak se provádí výpočet v homogenních souřadnicích.

- převod 3D vektoru na homogenní vektor
 - každý vektor (x, y, z) lze převést pomocí rovnice $(x, y, z, 1) \rightarrow$ vzniká 4D vektor x, y, z, w
- aplikace homogenní transformace
 - 4D homogenní vektor se transformuje pomocí homogenní transformace
 - matice 4x4
- převod zpět na 3D vektor
 - 4D vektor se převede pomocí vydělení všech složek hodnotou w
 - $x' = x / w$
 - $y' = y / w$
 - $z' = z / w$

Práce v homogenních souřadnicích



2. Vysvětlete princip rasterizace trojúhelníka.

a) Interpolace v ploše trojúhelníka

- každý pixel se vyplní barvou odpovídající interpolovaným hodnotám barev ve vrcholech trojúhelníku

pro trojúhelník:

rozdělíme trojúhelník vodorovně

(seřadit podle y)

$A_y \leq B_y \leq C_y$ prohazovat všechny souřadnice bodů

Pro y od A_y do B_y

lineárně interpolujeme podél hran souřadnici x podle y

$$s1 = (y - A_y) / (B_y - A_y)$$

$$s2 = (y - A_y) / (C_y - A_y)$$

$$x1 = A_x \cdot (1 - s1) + B_x \cdot s1$$

$$x2 = A_x \cdot (1 - s2) + C_x \cdot s2$$

pro dané y procházíme od x1 do x2 (nebo obráceně)

for (x = x1; x < x2; x++)

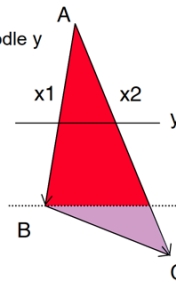
vypočítat z (interpolací)

vyřešit viditelnost

případně vykreslit PutPixel(x,y)

Pro y od B_y do C_y

to samé pro druhou polovinu



b) Interpolace na hranách

- při vykreslování hran trojúhelníka se interpolují hodnoty atributů v jeho vrcholech
- tyto interpolované hodnoty se používají pro vykreslování výplňových pixelů trojúhelníka
- postup:
 - výpočet atributů vrcholů
 - výpočet přírůstků atributů
 - interpolace atributů na hranách
 - výpočet atributu pro každý pixel

pro trojúhelník:

rozdělíme trojúhelník vodorovně

(seřadit podle y)

$A_y \leq B_y \leq C_y$ prohazovat všechny souřadnice bodů

Pro y od A_y do B_y

lineárně interpolujeme podél hran souřadnici x podle y

$$s1 = (y - A_y) / (B_y - A_y)$$

$$s2 = (y - A_y) / (C_y - A_y)$$

$$x1 = A_x \cdot (1 - s1) + B_x \cdot s1$$

$$x2 = A_x \cdot (1 - s2) + C_x \cdot s2$$

pro dané y procházíme od x1 do x2 (nebo obráceně)

for (x = x1; x < x2; x++)

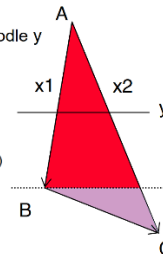
vypočítat z (interpolací)

vyřešit viditelnost

případně vykreslit PutPixel(x,y)

Pro y od B_y do C_y

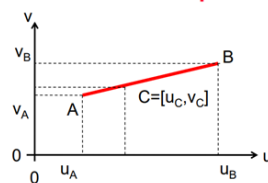
to samé pro druhou polovinu



3. Co je to interpolace, jak se využívá v zobrazovacím řetězci.

- proces výpočtu neznámých hodnot mezi známými hodnotami na základě jejich vzájemného vztahu
- slouží k dosažení plynulého přechodu nebo hladkých efektů mezi body

Lineární interpolace



$$t = \frac{(u_C - u_A)}{(u_B - u_A)}$$

„odečíst minimum,
dělit rozdílem“

$$v_C = (1 - t)v_A + tv_B$$

$$\begin{aligned} v_C &= v_A - tv_A + tv_B \\ &= (v_B - v_A)t + v_A \\ &= f(t) \end{aligned}$$

$$y = f(x) = kx + q$$

a) Interpolace souřadnic x, y, z

- proces vyhodnocování neznámých hodnot souřadnic mezi dvěma nebo více známými body v prostoru
- používá se pro určení polohy bodů na křivkách nebo plochách, které jsou popsány pouze jejich řídicími hodnotami

b) Interpolace atributů ve vrcholech

- používá se pro přenos vlastností mezi vrcholy
 - barva
 - textura
 - normála
- provádění se normální interpolace dehomogenizovaných atributů a následné násobení interpolovanou hodnotou w'

c) Princip perspektivně korektní interpolace

- zajištění správného vykreslení objektů v 3D prostoru, tak aby perspektiva byla zachována a předměty ve scéně měli realistický vzhled
- interpolace na základě hodnot ve vrcholech trojúhelníka a jejich vzdálenosti od pozorovatele
 - například při interpolaci z -hodnot při vykreslování 3D scény, kdy se musí brát v úvahu perspektivní zkreslení

4. Metody optimalizace vizualizace

- snaha o urychlení zobrazení scény v reálném čase

a) Úlohy vizualizace

- zobrazení složitých scén
 - volba vhodných datových struktur
 - předzpracování scény
 - využití hw prostředků
- vytváření realistických scén
 - nahrazení klasických algoritmů přibližnými metodami
- optimalizovat lze
 - řešení viditelnosti
 - ořezávání
 - kolize
 - osvětlení a stíny
 - určení viditelnosti bodu A z bodu B
- metody optimalizace
 - vhodná reprezentace objektů
 - redukce počtu zobrazovaných objektů
 - na úrovni polygonů
 - na úrovni sektorů, strukturalizace scény
 - snižování složitosti objektů

b) Double buffering

- odstranění blikání při vykreslení scény
- použití dvou ImageBufferů
 - první buffer je zobrazen
 - druhý je schován a použit pro překreslení scény

- po vykreslení jsou oba buffery prohozeny
- další buffery pro stereoskopickou projekci
 - GL_FRONT_LEFT
 - GL_FRONT_RIGHT
 - GL_BACK_LEFT
 - GL_BACK_RIGHT
- OpenGL vždy alespoň levý přední buffer

c) Reprezentace scény

- způsob, jakým jsou objekty v 3D scéně uloženy do paměti pro snadnou manipulaci a rychlé vykreslování
- scéna N trojúhelníků
 - velká paměťová náročnost
 - nutnost předání a zpracování velkého počtu vrcholů
 - Index buffer
- scéna N polygonů

d) Display list

- seznam příkazů pro vykreslování 3D scény uložené v paměti
 - pro rychlé a efektivní vykreslení scény
- v metodě init:
 - glNewList(jméno, parametr)
 - posloupnost příkazů
 - glEndList(jméno, parametr)
- v metodě display:
 - glCallList(číslo)

e) Backface culling

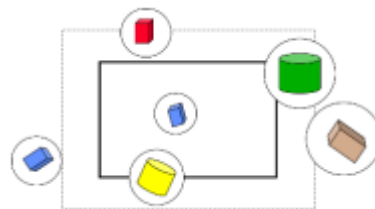
- odstranění odvrácených polygonů
 - odvrácené polygony nejsou zpracovány
 - není prováděna rasterizace, z-buffer....
 - orientace závisí na pořadí vrcholů, ne na normále
- glEnable(GL_CULL_FACE)
- glCullFace()
- glFrontFace()
- důležitá orientace polygonů / trojúhelníků

f) Frustum culling

- ořezání zobrazovacím objemem
- vyšetření ve 3D, homogenní souřadnice
 - u složitějších objektů výpočetně náročné
 - vytvoření obálek objektu – koule
 - detekce průsečíku koule a rovin omezujících zobrazovací objem
 - střed koule musí ležet uvnitř nebo vně ve vzdálenosti bližší než je poloměr

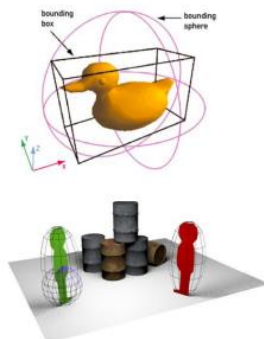


- vyšetření ve 2D
 - x, y v rozsahu od -1 do 1 v NDC, nebo po Viewport transformaci $\langle 0, \text{width} \rangle$ a $\langle 0, \text{height} \rangle$
 - vytvoření obálek objektů – kruh
 - promítnutí středu kruhu do průmětny + detekce uvnitř nebo vně
 - výpočet vzdálenosti středu od obdélníka
 - zvětšení obdélníka o velikost středu



g) Ohraničující objemy

- těleso obalující celý objekt – obálka
- jednoduchá geometrie
 - rychlý test viditelnosti / zobrazovacího objemu / kolize
 - základní geometrie tělesa
- rychlejší než provádění testů s plochami původního objektu
- typy
 - bounding sphere
 - bounding box
 - elipsoid
 - K-DOP
 - convex hull



h) Úroveň detailu

- metoda optimalizace, která snižuje úroveň detailu vzdálených objektů pro zrychlení vykreslení
- LOD (Level of Detail)

- contribution culling
 - některé objekty nezobrazovat
 - pokud je objekt příliš daleko / jeví se příliš malý, není zobrazen
- discrete LOD
 - různě složité objekty pro různé vzdálenosti
 - větší paměťová náročnost
 - nutnost vytvořit modely s různou složitostí
- continuous
 - automatická degradace objektu
 - hranový kolaps
 - obtížně řešitelné v obecném případě
- MIP Mapping
 - snižuje počet polygonů, nejčastěji například u stromu
 - výrazně zvyšuje výkon a snižuje náročnost na HW

i) Occlusion culling

- vyloučení elementů, které jsou překryty jinými pixely

j) Deferred shading

- metoda vykreslování, ve které je nejdříve renderována geometrie, poté je vypočítáno osvětlení a stínování pouze pro viditelné části scény
- lze velmi efektivně zpracovat scénu s mnoha světly
- nejprve se informace načítají z G-bufferu

k) Rozdělení scény

- Bounding Volume Hierarchy
 - vytvoření ohraničujících geometrií pro jednotlivé objekty
 - možnost překrývání objemů
- Potentially Visible Set
 - části scény, které mohou být vidět, ostatní není zpracováváno
 - je nutné řešit skutečnou viditelnost
 - příklady:
 - Portal Culling
 - možnost vzájemného propojení buněk pomocí portálů
 - pokud není vidět portál, nemůže být vidět buňka
 - BSP Tree
 - rekurzivní dělení prostoru vybranou obecnou rovinou
 - orientace normály určuje co je před a co za rovinou
 - k-D tree
 - osově orientovaný BSP
 - v každé úrovni stromu dělení podle roviny kolmé na jednu souřadnicovou osu
 - Octree
 - pravidelné dělení prostoru scény na 8 boxů (oktetů)
 - boxy jsou reprezentovány listy stromu

B. OpenGL

5. Knihovna OpenGL

a)základní principy

- multi-platformní knihovna pro 2D a 3D grafiku
- API ke grafické kartě
- pixelově orientované
- umí:
 - vykreslení grafických primitiv
 - transformace
 - mapování textur
 - výpočet osvětlení z bodových zdrojů
 - učení viditelnosti (z-buffer)
- neumí:
 - práci s okny
 - výpočet odrazů ve scéně
 - řešení vržených stínů
 - pokročilejší algoritmy musí být ručně provedeny v shaderech
- nástupce Vulkan

b)grafické elementy, atributy

- složeny z vrcholů
 - Vertex buffer
 - Index buffer
- modely složeny z:
 - bodů
 - čar
 - trojúhelníků
 - polygonů
- atributy:
 - velikost bodu
 - tloušťka čáry
 - typ čáry
 - způsob vykreslení
 - výplň polygonu
 - orientace vrcholů
 - ořezání
 - barva

6. Knihovna OpenGL: poskytované funkce pro zobrazení scény, řetězec zpracování

a)Modelování scény

- glBegin – začátek zadávání objektu
- glEnd – konec zadávání objektu
- vytvářejí se modely, které jsou složeny z množství geometrických primitiv
- modely se mohou transformovat rotací, posunem, nebo změnou velikosti

b)Řešení viditelnosti

- využití z-bufferu implementovanému na kartě

c) Osvětlení scény

- různé druhy světla v prostoru scény
- různé techniky osvětlení:
 - bodové
 - reflektové
 - směrové
 - plošné
- stínování:
 - GL_SMOOTH
 - Gouraudovo stínování
 - interpolace barvy ve vrcholech
 - GL_FLAT
 - konstantní stínování
 - obarveno barvou určitého vrcholu

7. Knihovna OpenGL: transformace

a) Způsob skládání transformací, násobení matic

- skládání transformací
- řádky krát sloupce
- Modelovací * Projekční * Pohledová
 - transformace vpravo se provedou jako první
- glPushMatrix(): uloží aktuální matici na zásobník
- glPopMatrix(): nahradí aktuální matici maticí na vrcholu zásobníku
- glMultMatrix(): násobí aktuální matici maticí, která je vložena jako parametr

b) Modelovací matice

- transformace aplikované na 3D objekty
- aplikují se na vertexy modelu
- translace, rotace, škálování
- glTranslate()
- glRotate()
- glScale()

c) Pohledová matice

- k určení, jak se bude scéna zobrazovat z určitého bodu
- transformace pohledové matice zahrnují rotaci a posun kamery

d) Projekční matice

- před viewport transformací
- používá se k transformaci 3D scény do 2D roviny
- od (-1, 1)
- transformace projekční matice zahrnují perspektivní projekci a určení pozice dálkové roviny

e) Zásobník transformací, použití

- operace
 - glPushMatrix()
 - uloží aktuální matici na zásobník
 - glPopMatrix()

- vybere matici ze zásobníku a aktualizuje s ní aktuální matici
- počet push a pop musí být stejný

f) Transformace do okna

- provádění se aby se určila velikost a pozice okna vykreslování
- `glViewport(x, y, w, h)`
 - transformace v rovině xy
- `glDepthRange(near, far)`
 - rozsah 0, 1
 - přepočítává Z souřadnici z intervalu od -1 do 1 (NDC) na definovaný interval
 - ovlivňuje přesnost z-bufferu

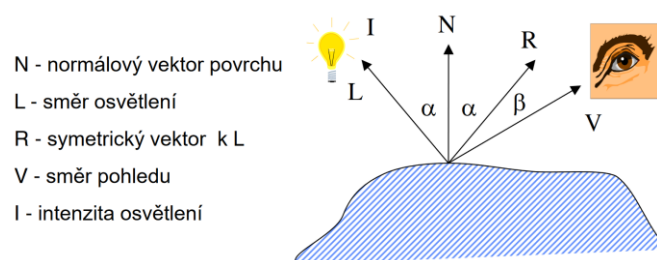
8. Knihovna OpenGL:

a) model osvětlení, typy zdrojů světla, vlastnosti

- různé techniky osvětlení:
 - bodové
 - všude přítomné světlo
 - paprsky jsou rovnoběžné
 - reflektové
 - všesměrový bodový zdroj
 - paprsky se rozbíhají
 - směrové
 - kužel světla z jednoho bodu
 - podobné jako bodové
 - plošné
 - vznikají polostíny
 - náročné na výpočet
- attenuation
 - zeslabení intenzity světla s rostoucí vzdáleností

b) model osvětlení, materiály

- phongův osvětlovací model



c) stínování a mlha

- snížení viditelnosti pozorovatele do dálky
- aplikuje se podle Z souřadnice
- parametry:
 - barva (`glFogfv`)
 - režim
 - lineární
 - nelineární

- hustota
 - startovací a koncová hodnota Z
- stínování:
 - GL_SMOOTH
 - Gouraudovo stínování
 - interpolace barvy ve vrcholech
 - GL_FLAT
 - konstantní stínování
 - obarveno barvou určitého vrcholu

d)textury, mapování a způsoby aplikace

- MIP mapování textur
 - různé rozlišení obrazu u textury v jednom souboru
 - vybrána textura vhodné úrovně podle vzdálenosti od pozorovatele
- Interpolace textury
 - mapování textury při zvětšení / zmenšení
 - velikost textury x velikost rastrového elementu
- glEnable(GL_TEXTURE_2D)
 - zapnutí textur
- glBindTexture(GL_TEXTURE_2D, textureID)
 - navázání textury na objekt
- glTexEnv(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, textureEnvMode)
 - nastavení způsobu aplikace textury
- glTexImage2D(GL_TEXTURE_2D, 0, GL_RGB, width, height, 0, GL_RGB, GL_UNSIGNED_BYTE, data)
 - načtení textury

e)míchání barev

- neboli blending
- míchání nového pixelu s tím, co už je umístěn v Image bufferu
- lineární kombinace barev
 - $src * srcFactor + dst * dstFactor$
- OpenGL
 - glEnable/Disable(GL_BLEND)
 - glBlendFunc(SourceFactor, DestinationFactor)
- využití:
 - částečně transparentní textury

C. Zpracování obrazu a počítačové vidění

9. Komprese rastrového obrazu principy, parametry, použití.

- transformace dat z jedné reprezentace do druhé
- zmenšení velikosti dat při zachování všech informací

a) Ztrátová komprese

- odstraňuje informace, které nejsou pro lidské oko významné
 - redukce barev
 - ztráta přesnosti geometrických tvarů
- vysoká komprese – 50:1
- JPEG

b) Bezeztrátová komprese

- přesně 100% zachovaný obraz
 - odstranění nevyužitých bitů
 - odstranění redundance kódu
- nízká komprese – 3:1

10. Komprese grafických dat: metody

a) LZW

- Lampel, Ziv, Welch
- substituční algoritmus
- slovníková metoda komprese
 - postupně rostoucí slovník vytvářen za běhu
 - délka kódu je kratší než délka kódovaného řetězce
 - rozšířená vstupní abeceda (12 bitů)
 - na výstupu pouze indexy do slovníku
 - slovník vytvářen pomocí kódování a dekódování
 - speciální znaky na mazání a vytváření slovníku
- jednoduchý algoritmus, složitá implementace
- odstranění šumu před kompresí
- GIF, TIFF, zip, arj

b) RLE

- Run Length Encoding
- kontextově závislá komprese
- opakující se hodnoty jsou nahrazeny počtem opakování a hodnotou
- JPEG, BMP, PCX

c) Huffmanovo kódování

- statická metoda komprese
 - zjišťuje četnost výskytu kódu
 - použití prefixového kódu
 - každá hodnota je reprezentována unikátní bitovou hodnotou
- nahrazuje běžné symboly nebo kombinace symbolů kratšími kódy
- komprese
 - spočítat výskyt všech znaků
 - seřadit podle počtu výskytů

- vytvořit binární strom
- pro každou hodnotu najít binární reprezentaci
- dekomprese
 - načíst počáteční hodnoty a vytvořit strom
 - čtení bitů ze souboru a procházení stromu
- JPEG, TIFF, FAX

11. Grafické formáty, použití, způsob komprese: GIF, TIFF, PCX, JPEG atd.

a) GIF

- pro ukládání obrazových souborů
- omezení 256 barev
- používá se především pro jednoduché obrázky a animace
- verze:
 - 87a
 - prokládání obrazu
 - umístění bloků na pozadí
 - 89a
 - doprovodný text vypisovaný na obrazovku
 - řízení grafiky
 - překreslování více obrázků
 - prodleva
 - průhlednost

b) TIFF

- formát pro ukládání rastrových obrazů
- podporuje 16-bitovou barevnou hloubku
- vysoká univerzálnost
 - různé kódování bytů
 - různé barevné modely
 - různé komprese (RLE, LZW, JPEG)
- různé kategorie datových položek
 - velikost
 - způsob reprezentace
 - vlastní data – pruhy
 - interpretace
- použití
 - DTP
 - zpracování obrazu

c) PCX

- pro ukládání rastrových obrazů
- původně pouze 16 barev – rozšířeno na 24 bitů
- komprese RLE

d) JPEG

- formát pro ukládání fotografických snímků s vysokým kompresním poměrem, ale určitou ztrátou kvality
- nastavení kvality – Q faktor

- snížení velikosti dat a zrychlení přenosového času
- komprese
 - transformace předlohy do optimálního barevného prostředí
 - redukce barev
 - odstranění redundantních dat
 - kvantizace DCZ koeficientů
 - cik-cak výběr výsledných koeficientů a aplikace Huffmanova kódování

e) PNG

- nástupce GIFu
 - nepodporuje animace
- určeno primárně pro internet
- bezztrátová komprese
- vlastnosti:
 - TrueColor – 24 bitů
 - detekce poškozených dat při přenosu
 - různé komprese
 - věrné zobrazení

f) ostatní

- TGA, SGI, JNG, MNG, DjVu, MrSID, DICOM, BPG, WebP, FLIF

12. Počítačové vidění a rozpoznávání, řetězec zpracování a rozpoznávání obrazu, aplikace

a) počítačové vidění

- obor zabývající se zpracováním obrazu pomocí PC
 - rozpoznání objektů v obraze
 - vyhledávání známých tvarů a entit
- používají se algoritmy a techniky zpracování signálů

b) řetězec zpracování

- proces identifikace
 - klasifikace a přiřazení významu obrazům
 - používá se především v oblasti umělé inteligence
- předzpracování
 - změna formátu, barev, ořezání...
- segmentace obrazu na objekty
- popis objektů

c) aplikace

- rozpoznání textu
- dálkový průzkum země
- lékařství
- sledování dopravy
- identifikace lidí
- pohyb robotů

13. Snímání obrazu

a) Požadavky na snímání, digitalizace a reprezentace obrazu

- kvalita obrazu, rozlišení a citlivost senzoru
- analogové signály se musí převést na digitální pomocí A/D převodníků
- digitální obraz může být reprezentován pomocí rastrových nebo vektorových dat

b) Vzorkování

- provedení digitalizace signálu v pravidelných intervalech
- čím vyšší vzorkovací frekvence, tím vyšší rozlišení obrazu, ale také větší datový objem
- typy
 - pravidelné
 - náhodné
 - roztřesené
 - semi-jitter
 - poissonovo diskové vzorkování

c) Kvantování

- reprezentace digitálního obrazu pomocí konečného počtu úrovní jasu
- čím větší počet úrovní, tím vyšší kvalita obrazu, ale větší datový objem
- typy
 - uniformní
 - konstantní délka intervalu
 - rovnoměrné rozložení
 - neuniformní
 - proměnná délka intervalu
 - méně využívaná
 - technicky náročnější

d) Snímání prostorové informace

- pokud senzor nezachycuje pouze intenzitu, ale také polohu, může vzniknout obraz s prostorovou informací
- typy
 - pasivní triangulace
 - jeden snímáný obraz
 - 1 kamera
 - stereoskopické vnímání – využití 2 pohledů z různých úhlů
 - aktivní triangulace
 - osvětluje scénu
 - dotykové snímání
 - laserové snímání
 - Motion Capture
 - snímání 2D obrazu
 - dírková kamera
 - všechno zaostřené
 - zachovává linie
 - kamera s čočkou
 - vzniká zakřivení
- expozice

- citlivost (ISO)
 - s citlivostí roste šum
 - Scanner – jednorozměrný senzor, který se postupně hýbe
 - CCD / CMOS
- šum
 - nadbytečná informace, která nesouvisí s obrazem
 - bílý šum
 - nadbytečné informace
 - sůl a pepř
 - hodnoty $\{-max, 0, max\}$
 - kvalita signálu
 - poměr signál-šum
- 3D snímání
 - LIDAR
 - výstupem jsou body umístěny v prostoru
 - vhodné pro snímání vegetace
 - Medicínské scannery
 - MRI

14. Předzpracování obrazu

a) Operace s obrazem

- aritmetické operace (sčítání, odčítání, násobení)
 - kombinace 2 obrazů
 - problém s přetečením rozsahu – saturace
- logické operace (AND, OR, XOR, NOT)
- konvoluce (hladké a ostřejší filtry)
- převod na odstíny šedi

b) Geometrické transformace obrazu

- přepočítání souřadnic obrazu na nové
 - posunutí, otočení, škálování, zkosení
- transformační funkce
- deformace
- warping
 - zmačkání, pokřivení
 - trojúhelníková síť
- morphing
 - spojitý přechod mezi 2 obrazy
- transformace rastru
 - mapování jednotlivých pixelů vstupního obrazu na výstupní rastr

c) Interpolace obrazu

- získání nových hodnot pixelů na základě existujících
- postupy:
 - nearest neighbor
 - získání hodnoty nejbližšího souseda (zaokrouhlení)
 - bilineární interpolace
 - interpolace na základě 4 okolních hodnot

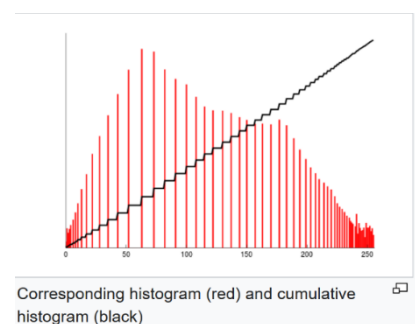
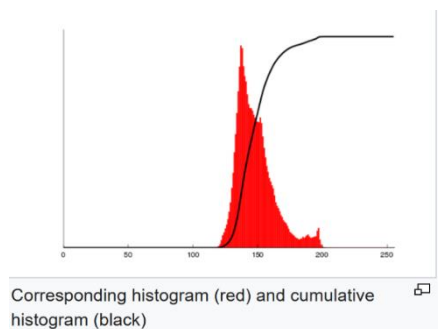
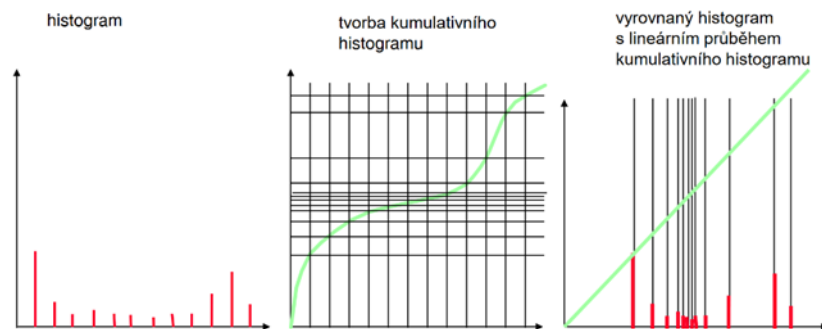
- bikubická interpolace
 - použití okolních 16 sousedů
- kosinová interpolace
 - použití 4 sousedů
- Lanczos interpolace
 - zachování detailu
 - malý alias při vzorkování

d)Úprava jasové funkce

- změna celkového kontrastu nebo jasu
- histogram obrazu lze použít k optimalizaci jasové funkce
- typy
 - roztažení kontrastu
 - prahování
 - vytváření monochromatického obrazu
 - gamma korekce
 - přizpůsobuje obraz lidskému oku nebo charakteristice snímáče

e)Histogram a vyrovnání histogramu

- ukazuje, kolik pixelů má určitou intenzitu
- globální informace
- vyrovnání histogramu
 - equalizace
 - nelineární přerozdělení úrovní kvantifikace, aby kumulativní histogram měl lineární průběh
 - nemění se četnosti ale úroveň intenzit



- vyhlazení histogramu
 - obsahuje řadu lokálních minim a maxim

- provedení vyhlazená váženým průměrem
- plovoucí okno s váženým průměrem – průměrování několika hodnot z okolí

15. Filtrace obrazu

a) Konvoluční filtry – aplikace, příklady

- matematická operace aplikovaná na pixely obrazu, kde nová hodnota pixelu je váženým průměrem jeho okolí
- používá se k vyhlazení obrazu, zvýraznění hran nebo odstranění šumu
- příklady
 - obyčejné průměrování
 - nový jas je průmětem okolních hodnot
 - Gaussův filtr
 - rozmazání pomocí normálního rozdělení
 - nelineární filtrace
 - filtrace pomocí mediánu

b) Gradientní operátory – použití, příklady

- operace, které vypočítávají gradient obrazu, tedy změnu intenzity v jednotlivých směrech
- používají se k detekci hran a změn v intenzitě obrazu
- typy
 - Robertsův operátor
 - velká citlivost na šum
 - nesymetrický
 - Sobelův operátor
 - aproximuje 1. derivaci, je směrově závislý
 - zvýhodňuje sousedy přes hrany (4-okolí)
 - používá se pro detekci hran
 - Cannyho hranový detektor
 - kritéria
 - žádná hrana nesmí být opomenuta
 - rozdíl mezi skutečnou a nalezenou hranou musí být minimální
 - potlačení hran s nemaximální hodnotou

c) Frekvenční filtry

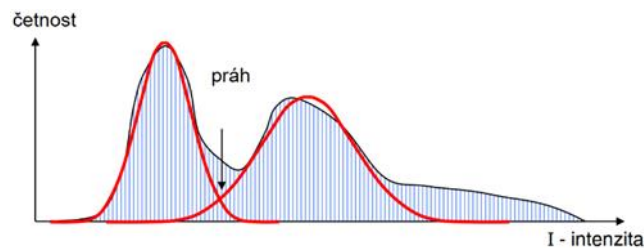
- používají se k odstranění periodického šumu nebo změn v intenzitě obrazu na určitých frekvencích
- Fourierova transformace
 - rozložení komplexního signálu na jednotlivé složky
- filtry
 - dolní propust
 - odstraňuje z obrazu složky vyšších frekvencí
 - odstraňuje detail a šum
 - průměrování
 - horní propust
 - odstraňuje složky s nízkou frekvencí
 - detekce bodů a hran
 - pásmová propust
 - propouští pouze vybrané složky frekvencí

- vyšší dimenze matice filtru
- navržení filtru pro konkrétní aplikace
- převod do frekvenční oblasti

16. Segmentace obrazu

a) Prahování, metody určení prahu

- rozděluje pixely do dvou skupin
 - hodnota přesahuje určitý práh
 - hodnota je pod prahovou hodnotou
- metodu určení prahu
 - pevné hodnoty prahu
 - pomocí známé kalibrované stupnice
 - podle fyzikálního principu snímání
 - procentní prahování
 - znalost procentuálního pokrytí objekty
 - analýza histogramu
 - podle počtu vrcholů
 - hodnota mezi vrcholy určuje hodnotu prahu

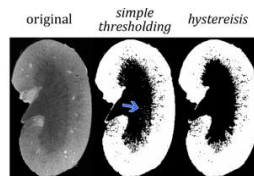


- otsuova metoda
 - statická metoda optimálního prahu
 - statická analýza histogramu
 - rozdělení na 2 třídy B a O s určitým prahem T
- iterační metoda
 - předpoklad pro první iteraci:
 - všechny pixely patří k oblasti Objektu kromě 4 rohových, které patří Pozadí
 - pro obě třídy určíme dělicí hodnotu použitou v další iteraci
 - další iterace:
 - spočítáme průměrnou hodnotu pro Objekty a Pozadí a zase spočítáme novou hodnotu prahu
 - v iteraci pokračujeme, dokud hodnota nekonverguje
- prahování barevného obrazu
 - převedeme na grayscale
 - nebo v každých složkách zvlášť
 - nebo převod do jiného barevného modelu

b) Segmentace na základě detekce hran, Houghova transformace

- spočívá v nalezení bodů v obraze, kde se jeho jas mění rychle
- heuristické sledování hranice
 - velikost hran tvořící hranici
 - vzdálenost od předpokládané hranice

- odhadu ceny z uzlu do koncového uzlu
- zjednodušení stromu řešení
- nejmenší maximální ceny
- hledání s omezením cesty
- hranové detektory
 - pixely označené jako hraniční se snažíme pospojovat do oblasti



- Houghova transformace
 - úloha nalezení daného předmětu v obraze
 - nutná znalost rovnice hraničních křivek
 - necitlivost výsledků na šum
 - pro všechny body obrazového prostoru lze nalézt obraz odpovídající všem možným přímkám procházejících obrazovým bodem
 - obrazem je opět přímka
 - průsečík přímek nám dává parametry hledané přímky

c) Segmentace na základě detekce oblastí, narůstání a štěpení oblastí, srovnávání se vzorem.

- informace o spojitých oblastech v obraze, které mají podobné vlastnosti
- homogenita
 - závisí na konkrétním segmentačním mechanismu, může být založena na odstínu šedi, tvaru, textuře...
 - výsledná oblast v segmentovaném obraze musí být homogenní a současně maximální
- region growing
 - záplavový algoritmus
 - počátek – semínko, oblast
 - hranice – kritérium homogeneity
 - sousednost – 4 a 8 konektivita
- přístupy
 - spojování oblastí
 - na začátku každý pixel jednotlivá oblast, postupně spojujeme
 - Pozor na 4 nebo 8 sousednost
 - štěpení oblastí
 - na začátku celý obrázek jedna oblast, postupně štěpíme
 - dělení může být pravidelné – např. kvadrantový strom
 - dělení může být i nepravidelné
 - segmentace srovnání se vzorem
 - testování souhlasu vzoru s obrazem
 - výpočet korelace
 - úroveň celých objektů
 - úroveň malých vzorů
 - problémy
 - geometrická transformace objektu v obraze
 - přítomnost šumu

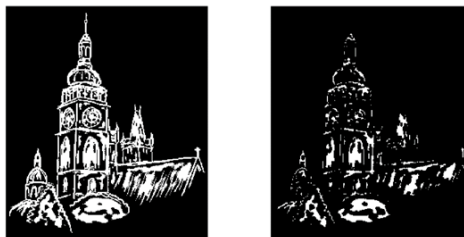
- brute-force algoritmus
 - snažíme se namapovat vzor (s různými transformacemi) na obrázek
 - pro všechny možné transformace určit míru vzoru a obrazu

d)Matematická morfologie – operace, použití

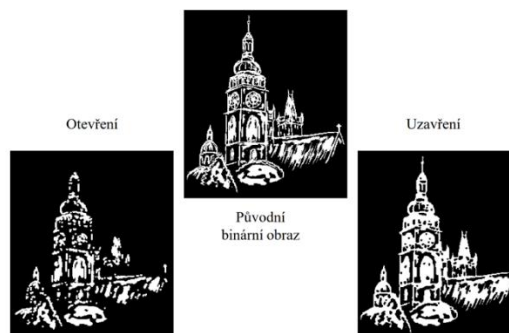
- metoda pro analýzu tvaru a struktury obrazu pomocí matematických operací jako eroze a dilatace
- nauka o tvarech
- dilatace
 - objekty se rozrůstají podle velikosti strukturního elementu
 - zaplnění malých děr a úzkých zálivů



- eroze
 - zjednodušení struktury objektu
 - rozklad objektů spojených čarami na několik objektů
 - nalezení obrysů objektů



- otevření a uzavření
 - používá se pro odstranění detailů menších, než je velikost strukturálního elementu
 - otevření
 - oddělí objekty spojené úzkou šíjí
 - zjednoduší strukturu objektů
 - uzavření
 - spojuje blízké objekty
 - zaplní malé díry a vyhladí obrys



17. Popis objektů a porozumění obrazu, klasifikace, kvantitativní charakteristiky, obarvení obrazu.

a) popis objektů a porozumění obrazu

- interpretace obrazu, klasifikace oblastí
- teorie rozpoznávání, umělá inteligence, statistika
- třídy objektů
- klasifikátor, kritéria
- popis objektů
- obrazový prostor

b) klasifikace

- rozlišování objektů do skupin na základě rysů
- použití strojového učení k naučení se klasifikovat objekty

c) kvantitativní charakteristiky

- vlastnosti objektů vyjádřené kvantitativně - např. velikost, tvar, barva, textura
- využití statistických metod pro analýzu a porovnání charakteristik objektů

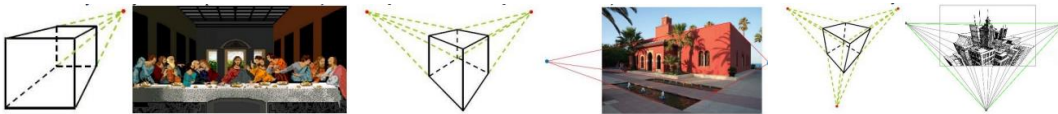
d) obarvení obrazu

- změna barev v obrazu pro zlepšení vizuálního dojmu nebo zvýraznění určitých rysů
- aplikace barevných filtrů pro zvýraznění konkrétních oblastí v obrazu

D. Doplnkové

1. Metody znázornění 3D prostoru

- projekce (promítnutí)
- stereoskopie
 - vytvoříme 2 pohledy na scénu
 - pro pravé a levé oko
 - výsledný obraz vnímáme ve 3D
- osvětlení a vržené stíny, barvy, textury a jiné efekty
 - pomohou nám rozpoznat, co vlastně vidíme
 - dodají výslednému obrazu atmosféru nebo styl
- pohyb (animace)
- konkrétní způsoby projekce
 - paralelní promítání
 - promítnutí do některé z rovin (x, y, z)
 - perspektivní
 - odpovídá lidskému vidění světa
 - typy
 - jednoúběžníková perspektiva
 - dvouúběžníková perspektiva
 - trojúběžníková perspektiva



2. Jak popisujeme transformace v 2D.

- proces, při kterém dochází ke změně polohy, orientace, velikosti
- lineární transformace
 - posunutí
 - otočení
 - změna měřítka
 - zrcadlení
 - zkosení
- obrazem bodu je bod
- vektorová a rastrová grafika
 - vektor – transformace se aplikuje na všechny řídicí body
 - rastr – aplikuje se na všechny pixely rastru

3. Jaké základní lineární transformace v 2D používáme, jak jsou definovány

- Posunutí: $X' = X + Px, Y' = Y + Py$
- Změna měřítka: $X' = X * Zx, Y' = Y * Zy$
- Otočení: $X' = X * \cos(a) - Y * \sin(a)$
 $Y' = X * \sin(a) + Y * \cos(a)$
- Zkosení, zrcadlení, bodová symetrie

Maticy

Změna měřítka

$$S(Sx, Sy) = \begin{pmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Pokud je $Sx = Sy$, je možné se stejným efektem použít matici

$$S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & Sx \end{pmatrix}$$

Tzn. nastavením homogenní souřadnice lze dosáhnout změny

Rotace

$$R(\alpha) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Posunutí

$$T = \begin{pmatrix} 1 & 0 & Px \\ 0 & 1 & Py \\ 0 & 0 & 1 \end{pmatrix}$$

Zkosení (Shear)

$$S = \begin{pmatrix} 1 & Zx & 0 \\ Zy & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$Sh_x(sh_x) = \begin{bmatrix} 1 & sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad Sh_y(sh_y) = \begin{bmatrix} 1 & 0 & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

4. Jak popisujeme transformace v 3D.

- transformace světa
 - změna měřítka, translace, rotace objektů
 - transformace může být výsledkem součinu několika transformačních matic
 - transformace světa jsou většinou jedinečné pro každý 3D objekt ve scéně
- pohledová transformace
 - tato transformace se používá pro úpravu celé scény
 - transformace upraví všechny vrcholy a vyjádří je v nových lokálních souřadnicích
- projekční transformace
 - může přidat perspektivní nebo ortogonální projekci
 - použití homogenních souřadnic
 - nutné před vykreslením homogenizovat

5. Jaké základní lineární transformace v 3D používáme, jak jsou definovány.

- Posunutí

Posunutí ve 3D je určeno vektorem posunutí $\vec{p} = (X_t, Y_t, Z_t)$. Transformační matice posunutí \mathbf{T} a inverzní matice \mathbf{T}^{-1} jsou

$$\mathbf{T} = \mathbf{T}(X_t, Y_t, Z_t) = \begin{bmatrix} 1 & 0 & 0 & X_t \\ 0 & 1 & 0 & Y_t \\ 0 & 0 & 1 & Z_t \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Změna měřítka

$$\mathbf{S}(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Rotace kolem os:

- X

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{Transponovaná:} \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Y

$$\begin{pmatrix} \cos\alpha & 0 & \sin\alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{Transponovaná:} \quad \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha & 0 \\ 0 & 1 & 0 & 0 \\ \sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Z

$$\begin{pmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{Transponovaná:} \quad \begin{pmatrix} \cos\alpha & \sin\alpha & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Zkosení:

$$\mathbf{Sh}_{yz} = \begin{bmatrix} 1 & sh_y & sh_z & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{Sh}_{xz} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ sh_x & 1 & sh_z & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{Sh}_{xy} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ sh_x & sh_y & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Tyhle matice zkosení ověřit (podle přednášek? – v učebnici nejsou):

$$\begin{pmatrix} 1 & a & b & 0 \\ c & 1 & d & 0 \\ e & f & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{Transponovaná:} \quad \begin{pmatrix} 1 & c & e & 0 \\ a & 1 & f & 0 \\ b & d & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

6. Jak funguje skládání transformací.

- Skládání transformací (matic) není komutativní = **záleží na pořadí prováděných transformací!**
 $\mathbf{A} \cdot \mathbf{B} \neq \mathbf{B} \cdot \mathbf{A}$

- Ale naštěstí je asociativní = **můžeme libovolně skládat do závorek**
 $\mathbf{A} \cdot (\mathbf{B} \cdot \mathbf{C}) = (\mathbf{A} \cdot \mathbf{B}) \cdot \mathbf{C}$

- Taky platí, že Maticové násobení je distributivní vůči sčítání = **lze vytýkat před závorku**
 $\mathbf{A} \cdot (\mathbf{B} + \mathbf{C}) = \mathbf{A} \cdot \mathbf{B} + \mathbf{A} \cdot \mathbf{C}$

- Přínásobením jednotkové matice E zleva i zprava se nic nezmění
 $\mathbf{E} \cdot \mathbf{A} = \mathbf{A} \cdot \mathbf{E} = \mathbf{A}$

- Transpozice součinu matic je součin transponovaných matic v opačném pořadí
 $(\mathbf{A} \cdot \mathbf{B})^T = \mathbf{B}^T \cdot \mathbf{A}^T$

- Inverzní matice součinu regulárních matic je součin inverzních matic v opačném pořadí
 $(\mathbf{A} \cdot \mathbf{B})^{-1} = \mathbf{B}^{-1} \cdot \mathbf{A}^{-1}$

- Řádkový vektor * matice = řádkový vektor
 $(\dots \dots \dots) \cdot \begin{pmatrix} \dots & \dots & \dots \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{pmatrix} = (\dots \dots \dots)$

- Sloupcový vektor * matice = sloupcový vektor
 $\begin{pmatrix} \dots \\ \dots \\ \dots \end{pmatrix} \cdot \begin{pmatrix} \dots & \dots & \dots \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{pmatrix} = \begin{pmatrix} \dots \\ \dots \\ \dots \end{pmatrix}$

- model
 - modelovací matice; vznikla složením matice rotace, posunu, měřítka, atd....
- view

- pohledová matice; vznikla složením vektorů popisujících pozorovatele
- projection
 - projekční matice (promítnutí); popisuje druh pohledu (perspektiva/ortogonální)

7. Kumulativní histogram

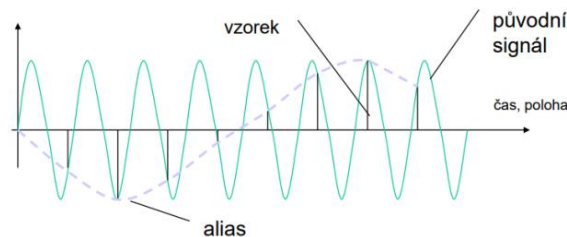
- [viz 14\) e\)](#)

8. Houghova transformace

- [viz 16\) b\)](#)

9. Co je to alias, kdy vzniká

- chyba při vzorkování spojité funkce



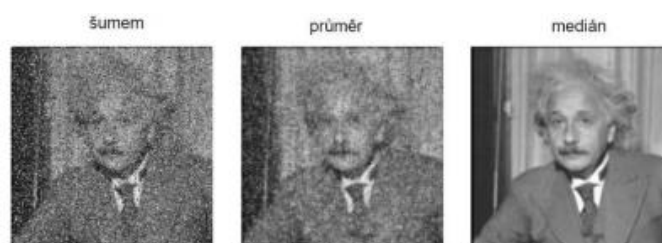
- časový alias
 - blikání
 - interference cyklického pohybu se snímkovou frekvencí
- prostorový alias
 - schodovité zobrazení šikmých čar



- interference obrazu s rastrem
- lze vyřešit anti-aliasingem

10. Mediánový filtr – použití

- nelineární metoda odstranění šumu
- vhodný k odstranění náhodného šumu
- algoritmus
 - pro každý pixel:
 - seřaď okolní pixely podle jejich intenzity
 - z těch pixelů urči medián a nahraď jím hodnotu zpracovávaného pixelu



11. Výpočet osvětlení ve scéně

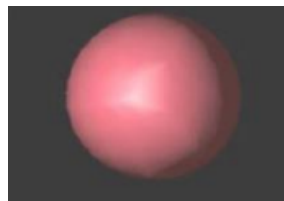
a. Vysvětlete pojem konstantní stínování.

- flat shading
- předpokládá se, že plocha má pouze jednu normálu
- barva závisí na normále plochy, která je stejná pro celou plochu
- tělesa s interpolovanými povrchovými plochami se jeví hranatá
- zvýraznění umělých hran
- vhodné pro hranatá tělesa



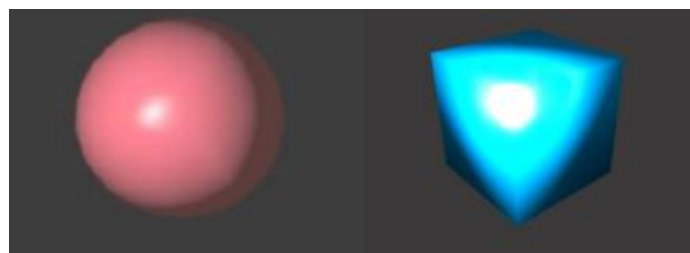
b. Vysvětlete pojem interpolace barvy.

- výpočet barvy mezi 2 jinými barvami
- také se nazývá Gouraudovo stínování
- osvětlení se vypočítá ve vrcholech plochy a vypočtená barva se interpoluje v ploše
- plynulé stínování



c. Vysvětlete pojem interpolace normály.

- vhodné v phongově stínování
- určena k plynulému stínování těles, jejichž povrch je tvořen množinou rovinných bodů
- vychází ze znalosti normálových vektorů ve vrcholech
- vyšší časová náročnost



d. Phongův osvětlovací model – ambientní složka.

- okolní světlo udává intenzitu té části světla, která na těleso dopadá rovnoměrně ze všech směrů
- konstantní osvětlení
- okolní světlo zajišťuje, aby povrchy odvrácené od světla nebyly zcela černé

e. Phongův osvětlovací model – difúzní složka.

- udává intenzitu té části světla, která se od matného povrchu tělesa rovnoměrně odráží do všech směrů
- množství odraženého světla závisí na směru dopadu světla
- čím více dopadne, tím více se odrazí
- tento jev popisuje Lambertův zákon

f. Phongův osvětlovací model – zrcadlová složka.

- udává intenzitu té části, která se od tělesa odráží převážně v jednom směru podle zákona odrazu