

POČÍTAČOVÁ GRAFIKA

PREDMET

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

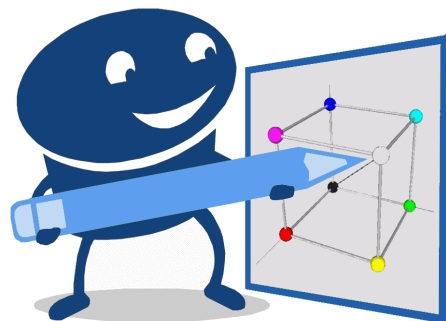
P 00

© 2025

CIEĽ PREDMETU

Predmet je zameraný na základy počítačovej grafiky z hľadiska potrieb infromatických odborov.

Cieľom je získať poznatky o princípoch práce grafických systémov s tým, že v niektorých prednáškach je venovaná pozornosť aj odvodeniam jednotlivých algoritmov.



PEDAGÓGOVIA

- doc. Ing. Branislav Sobota, PhD.
branislav.sobota@tuke.sk
- Ing. Timotej Sobota
timotej.sobota@tuke.sk



OBSAH PREDMETU

- Počítačová grafika – história, pojmy, súvislosti, aplikácie a súvisiace technológie
- Videnie, vizualizácia a vrstvy vizualizačného procesu
- Priestor, svetlo a farby
- Grafická informácia a grafické primitíva
- Priestor a objekty
- Transformácie grafických objektov
- Projekcie (premietacie transformácie)
- Krivky a plochy v počítačovej grafike
- Riešenie viditeľnosti
- Vyplňovanie a textúrovanie
- Tieňovanie, osvetľovanie a fotorealistické zobrazovanie
- Špeciálne oblasti PG



ZÁPOČET A SKÚŠKA

- **Zápočet:** Študent prospeje v PH a získa zápočet, keď splní podmienku získať min. 16b z 30b (Nadpolovičná väčšina bodov u posledných dvoch častí je podmienkou úspešného získania zápočtu).
 - Aktívne sa zúčastňuje výučby (prezenčne príp. online)
 - Úspešne absolvuje zápočtový test
 - Vypracuje semestrálny projekt
- **Skúška:** Študent prospeje v ZH a úspešne vykoná skúšku, keď splní podmienku získať min. 36b z 70b (Nadpolovičná väčšina bodov u oboch foriem je podmienkou úspešného absolvovania skúšky).
 - Forma skúšky: písomná (35%) a ústna (65%)
- **Celkové hodnotenie:** CH je suma hodnotení získaných študentom za hodnotené obdobie. Celkový výsledok sa stanoví v súlade s vnútornými predpismi TUKE. (študijný poriadok)

HODNOTENIE ZÁPOČTU A SKÚŠKY

- **Zápočet (max):** 30 bodov (nadpolovičná väčšina bodov u posledných dvoch častí je podmienkou úspešného splnenia) :
 - 10% (3 body) - dochádzka a aktívna účasť na cvičeniach
 - 40% (12 bodov) - zápočtový test
 - 50% (15 bodov) - semestrálny projekt, odovzdáva sa podľa pokynov cvičiaceho:
 - 40% (6 bodov) - funkcionálnosť,
 - 20% (3 body) - grafický vzhľad,
 - 20% (3 body) - dodržanie témy,
 - 20% (3 body) - doplnky, originalita, vylepšenia.
- **Skúška (max):** 70 bodov (nadpolovičná väčšina u oboch častí (23 a 13 bodov) je podmienkou úspešného absolvovania):
 - 65% (45 bodov) – ústna časť
 - 35% (25 bodov) – písomná časť - finálny semestrálny projekt vrátane dokumentácie, odovzdáva sa podľa pokynov prednášajúceho:
 - 24% (6 bodov) - funkcionálnosť,
 - 24% (6 bodov) - implementácia, kód,
 - 16% (4 body) - grafický vzhľad,
 - 12% (3 body) - dodržanie témy,
 - 12% (3 body) - doplnky, originalita, vylepšenia
 - 12% (3 body) – dokumentácia.

ODPORÚČANÉ ZDROJE

- Sobota, B.: **Počítačová grafika - podklady k prednáškam '24** - 1. vyd. - Košice : Technická univerzita v Košiciach - 2024. - 354 s.. - ISBN 978-80-553-4691-5.
- Sobota B., Hudák M., Mattoová M.: **Počítačová grafika - Návodý na cvičenia** - 2. vyd. - Košice : Technická univerzita v Košiciach - 2023. - 208 s. - ISBN 978-80-553-4417-1 .
- <https://moodle.fei.tuke.sk>
 - Počítačová grafika 1 (pg1)



LABORATÓRIUM LIRKIS

Unikátne pracovisko na výskum a vývoj nových flexibilných a inteligentných rozhraní systémov na báze počítačovej grafiky a technológií virtuálnej reality



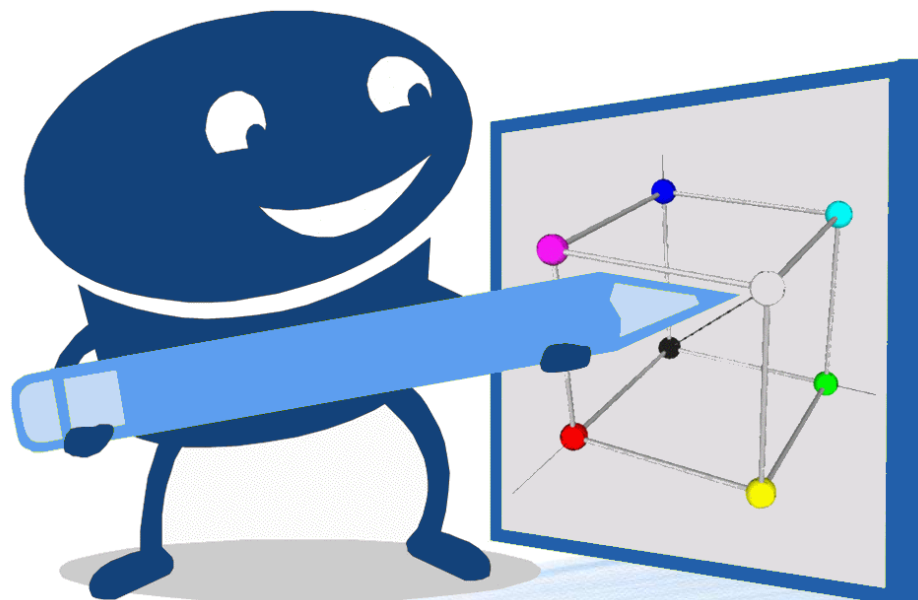
Laboratórium LIRKIS

Laboratórium Inteligentných Rozhraní

Komunikačných a Informačných Systémov

KPI FEI TU Košice

POČÍTAČOVÁ GRAFIKA



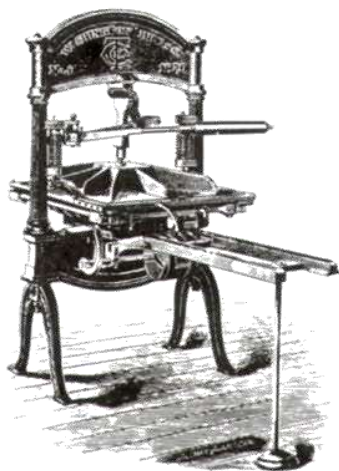
POČÍTAČOVÁ GRAFIKA - UPLATNENIE

- Forma komunikácie s počítačom a používateľské rozhranie
- Obaly v obchodoch
- Design automobilov či výrobkov spotrebnej elektroniky
- Projektovanie budov a interiérov
- Noviny, časopisy, katalógy
- Predpoveď počasia
- Počítačové hry a zábava
- Spracovanie hospodárskych či štatistických výsledkov
- Film, video, reklama

HISTÓRIA

- 1444-1448 - Johannes Guttenberg – kníhtlač
- 1805-1808 - Joseph Marie Jacquard zavádza tkáčsky stav riadený diernymi štítkami
- 1943 Alan M. Turing zostrojuje elektronický počítač COLLOSSUS
- 1946 John William Mauchly a John Presper Eckert stavajú univerzálny elektronický (na báze elektróniek) počítač ENIAC
- 1951 Ivan Moscowich stavia kresliaci stroj
- 1960 pracovníci firmy Boeing zavádzajú pojem "počítačová grafika,,
- 1961 Edward E. Zajac vytvára prvý film animovaný počítačom s názvom "Two-Gyro Gravity - Gradient Attitude Control System,,
- 1962 Ivan Sutherland predstavuje prvý interaktívny grafický program Sketchpad ovládaný svetelným perom s výstupom na grafickú obrazovku
- 1963 je vyrobený prvý "ploter" - kresliaci stroj riadený číslicovým počítačom

HISTÓRIA



1. kníhtlačiarenský stroj



aut. tkáčsky stav

1. myš



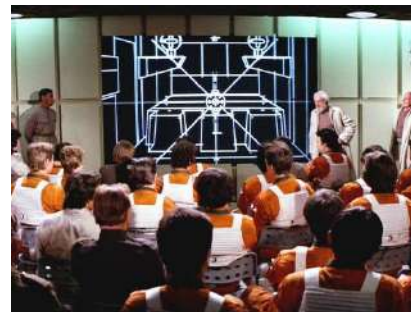
digigraf



práca s programom Sketchpad



Xerox Alto



grafika
v Star Wars



Apple macintosh

VIDENIE (VISION)

je komplementárny problém, ktorý analyzuje obrazy na vytvorenie počítačových modelov sveta



VYKRESĽOVANIE (RENDERING)

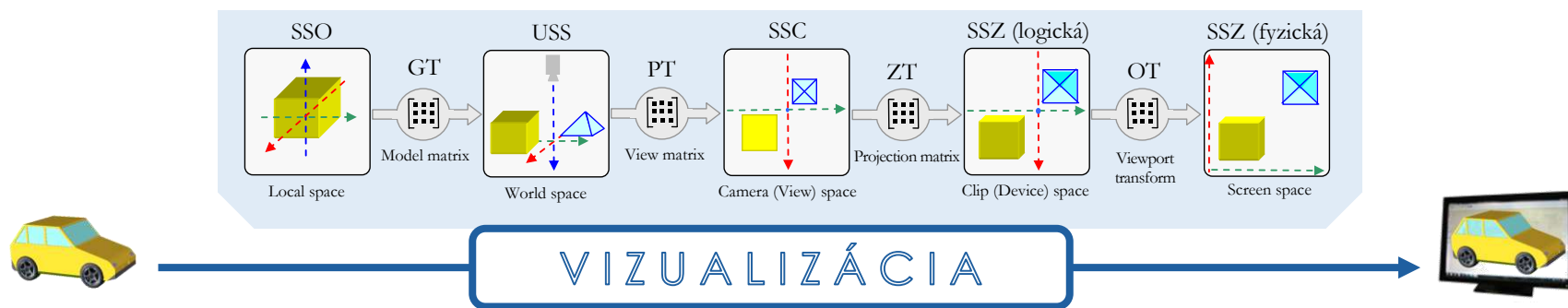
je proces tvorby obrazu pomocou počítačových algoritmov alebo tiež oblasť štúdia zaoberajúceho sa syntézou obrazov z modelov sveta v počítači.

VIZUALIZÁCIA

je proces transformácie popisu modelu virtuálneho sveta do výstupného obrazu na zobrazovacom zariadení

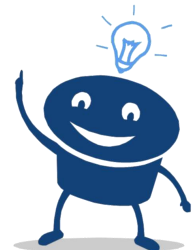


Základnou úlohou grafických systémov je generovať grafický výstup pre pripojené zobrazovacie zariadenie



VRSTVY VIZUALIZAČNÉHO PROCESU

1. Definovanie/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami (geometrické)
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

PRIESTOR, SVETLO A FARBY

doc. Ing. Branislav Sobota, PhD.

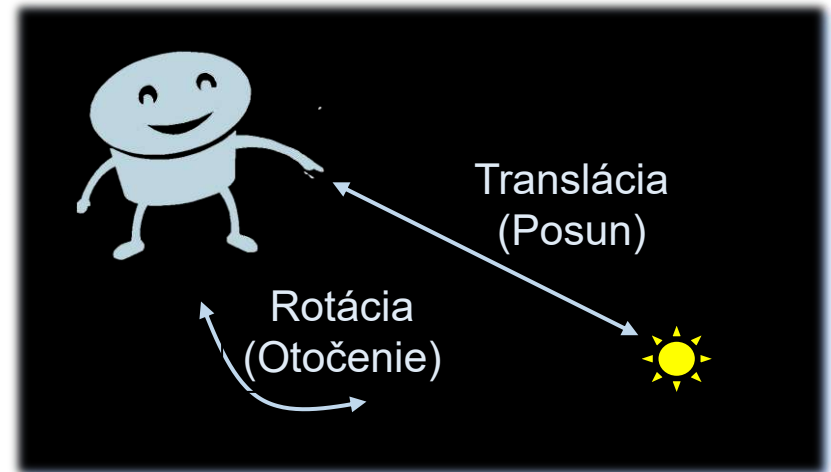
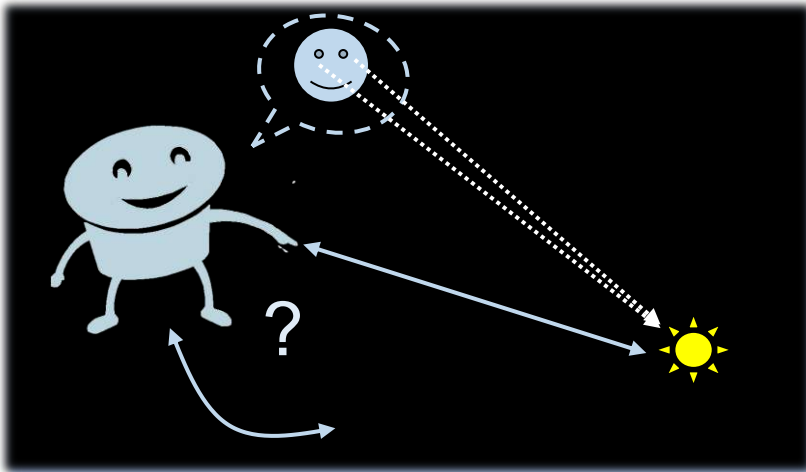
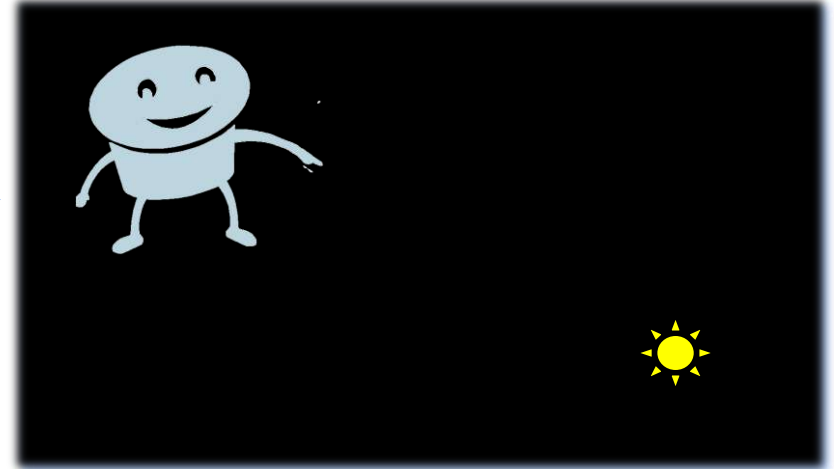
Katedra počítačov a informatiky, FEI TU v Košiciach

P 01

© 2025

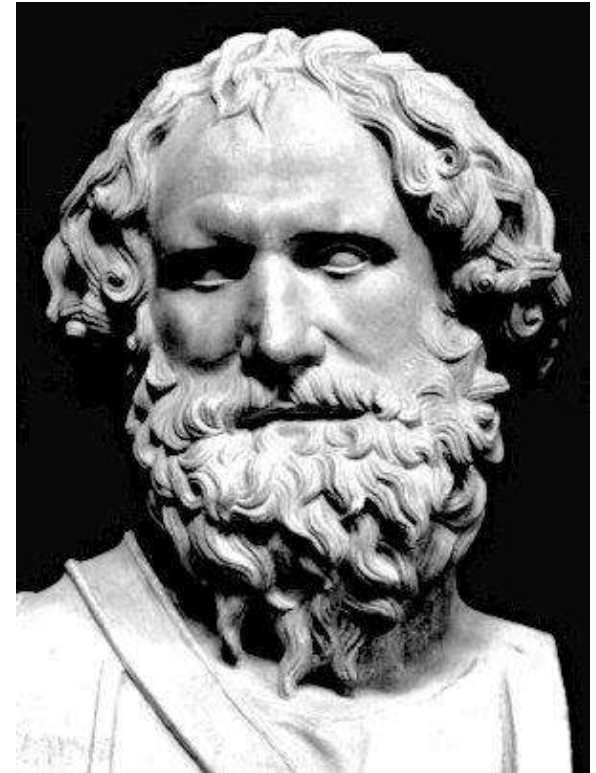
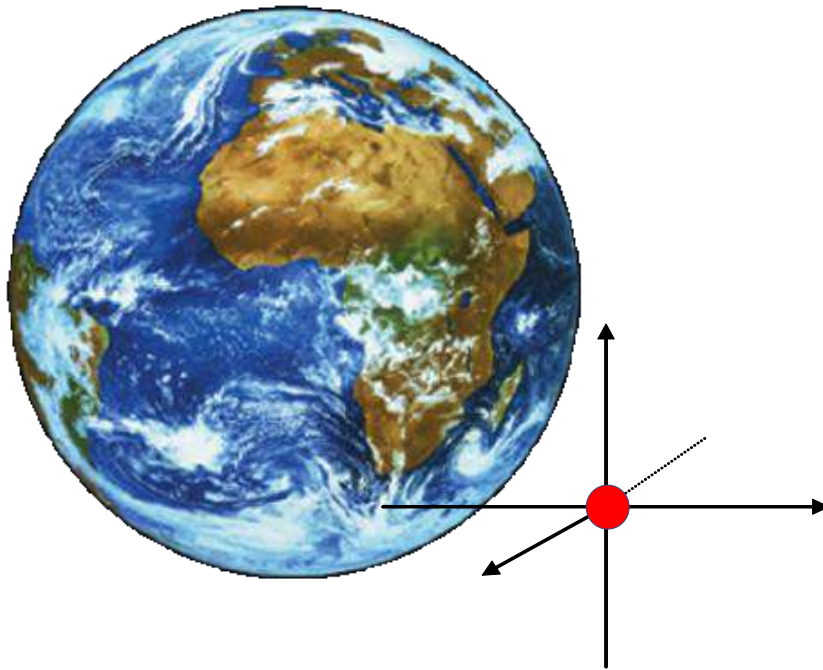
PRIESTOR





PRIESTOR A SÚSTAVY

„Dajte mi pevný bod a ja pohnem svetom.“



Archimedes
287-212 p.n.l.
Syrakúzy

DIMENZIA (ROZMER) PRIESTORU

Dimenzia je založená spravidla na *fyzikálnej veličine* (jednotke, primárne merateľnej) reprezentujúcu daný priestor alebo jeho zložku. Najčastejšie sa používajú základné a prípadne doplnkové alebo odvodené fyzikálne jednotky (dĺžka=meter, čas=sekunda a pod.). Koncept dimenzie nie je obmedzený na fyzické priestory. Vo fyzike a matematike je rozmer matematického priestoru (alebo objektu) neformálne definovaný ako minimálny počet parametrov (koordinát, súradníc) potrebných na špecifikáciu akéhokoľvek bodu v ňom. Vysokorozmerné priestory často vyskytujúce sa v matematike a prírodných vedách môžu byť priestory parametrov alebo konfiguračné priestory, ako napríklad v Lagrangovej alebo Hamiltonovej mechanike, sú to abstraktné priestory, nezávislé od fyzického priestoru, v ktorom žijeme.

Typy dimenzií:

- Číselné
- Nečíselné

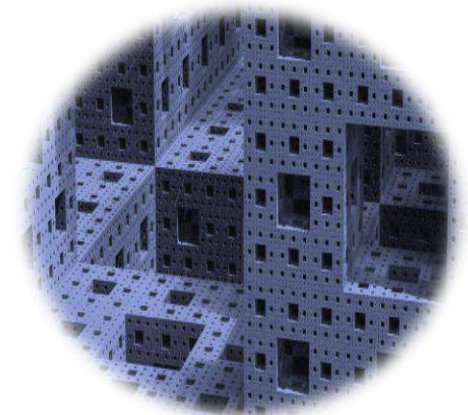
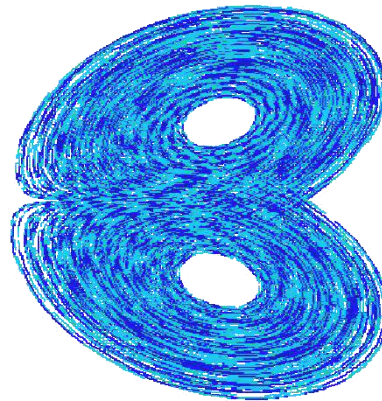
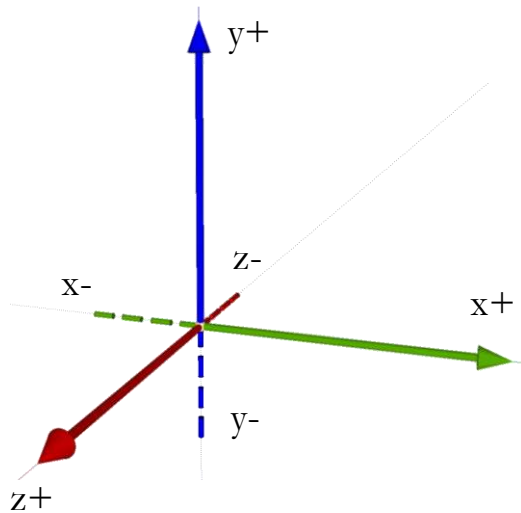
Druhy dimenzií:

- Topologická
- Hausdorffova
- Farebná
- Fraktálna
- dimenzia sebedobnosti
- Kapacitná
- Informačná
- dimenzia Rotácií
- ...



ČÍSELNÁ DIMENZIA (ROZMER) PRIESTORU

- **Celočíselná** (topologická) dimenzia (0,1,2,3,4), celočíselná metrika
- **Neceločíselná** dimenzia (2.6 a pod.)



ŠTRUKTÚRA DIMENZIE (ROZMERU) PRIESTORU

$$D(imenzia) = M + N + \dots$$

veličina 1
veličina 2
veličina n



Príklad štruktúry 3 hodnotovej dimenzie:

$3 = 3 + 0$ (geometria + čas) – *homogénna štruktúra*

$3 = 2 + 1$ (geometria + čas) – *nehomogénna (heterogénna) štruktúra*



PRIESTOR A JEHO SÚRADNICOVÁ SÚSTAVA

KOORDINAČNÝ SYSTÉM

- *Súradnicová sústava* (koordinačný systém) umožňuje parametrizovať priestor a definovať jeho počiatočný bod (stred súradnicovej sústavy, origin) a smery rozvoja fyzikálnej veličiny v príslušnej dimenzii priestoru popisovaného koordinačným systémom (jeho osi).
- *Súradnice* (parametre, koordináty) definujú jednoznačne polohu v rámci súradnicovej sústavy podľa charakteru a štruktúry dimenzií priestoru

Tak je položený základ systematickému popisu pohybu a orientácie v akomkoľvek priestore

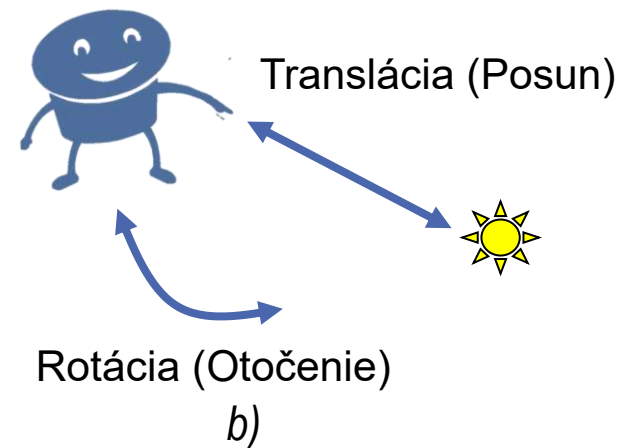
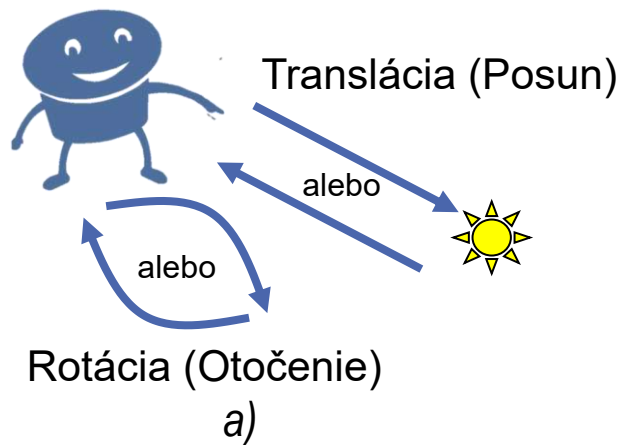


STUPEŇ VOĽNOSTI DIMENZIE

DEGREE OF FREEDOM – DOF

Stupeň voľnosti (Degree of Freedom – DOF) dimenzie je počet nezávislých parametrov, ktoré definujú konfiguráciu priestoru/objektu alebo stav v príslušnom koordinačnom systéme. DOF definuje počet translačných (T) a rotačných (R) (vrátane napr. času v heterogénnych štruktúrach) smerov, v ktorých sa môže objekt pohybovať:

- a) *Jednosmerné* vzhľadom na dimenziu (len v jednom smere dimenzie)
- b) *Obojsmerné* vzhľadom na dimenziu (v oboch smeroch dimenzie)



STUPEŇ VOĽNOSTI V 3D PRIESTORE

Poloha n -rozmerného tuhého telesa je definovaná tuhou transformáciou:

$$[T] = [A, d]$$

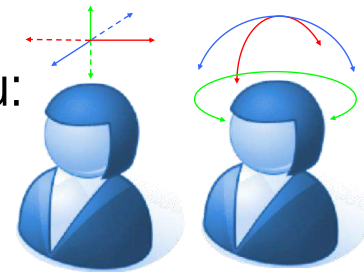
kde d je n -rozmerná translácia a A je matica rotácie $n \times n$.

Stupeň voľnosti systému možno považovať za minimálny počet súradníc potrebných na špecifikáciu konfigurácie. Aplikovaním tejto definície máme (pri obojsmernom systéme DOF):

- Pre jednu časticu v 2D priestore dve súradnice definujú jej polohu, takže má dva stupne voľnosti
- Jedna častica v 3D priestore vyžaduje tri súradnice, takže má tri stupne voľnosti
- Dve častice v priestore majú spolu šesť stupňov voľnosti.

Jedno pevné teleso má najviac šesť stupňov voľnosti (6 DOF (12 smerov jednosmerne)) 3T3R pozostávajúce z troch posunov (3T) a troch rotácií (3R).

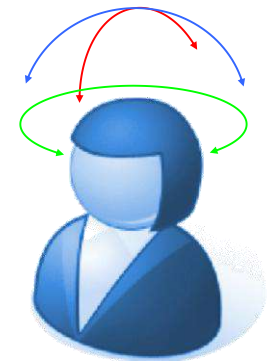
Poznámka: v robotických systémoch sa DOF definuje pre každý kĺb



3T

3R

3DOF vs. 6DOF



3T3R

POPIS A REPREZENTÁCIA PRIESTOROV

Pri spracovaní priestorov sú zaujímavé dve hľadiská:

- popis priestoru (ČÍM, napr. vektor RGB[35, 177, 89])
- reprezentácia priestoru (AKO (formát), napr. RGB kocka, HLS kužeľ a pod.)



DUALITA PRIESTOROV

Princíp duality priestorov hovorí, že každý výrok z hľadiska prvého priestoru zostáva pravdivý, keď je možné zameniť dva rôzne objekty v oboch priestoroch tak aby boli sémanticky rovnaké (napr. slová „bod“ a „priamka“, „ležať na“ a „prechádzať“, „spojiť sa“ a „priesečník“, atď.) a tak sa získa duálny výrok v druhom priestore.

Inými slovami, princíp duality hovorí, že vo všetkých výrokoch je možné napr. zameniť pojem „bod“ pojmom „priamka“ a pojem „priamka“ pojmom „bod“ atď. a daný výrok zostáva v platnosti. Podobná dualita platí aj pre svetelné priestory alebo farebné modely napr. na vyjadrenie farby. V praxi to znamená, že tú istú farbu je možné v rôznych farebných priestoroch opísať rôzne a použiť výhodnejší spôsob.



NÁUKY O SVETLE A FARBÁCH

- *Optika* – náuka o svetle a jeho vlastnostiach, svetelných javoch a prístrojoch
- *Chromatika* - náuka o farbách



SVETLO A SVETELNÝ PRIESTOR

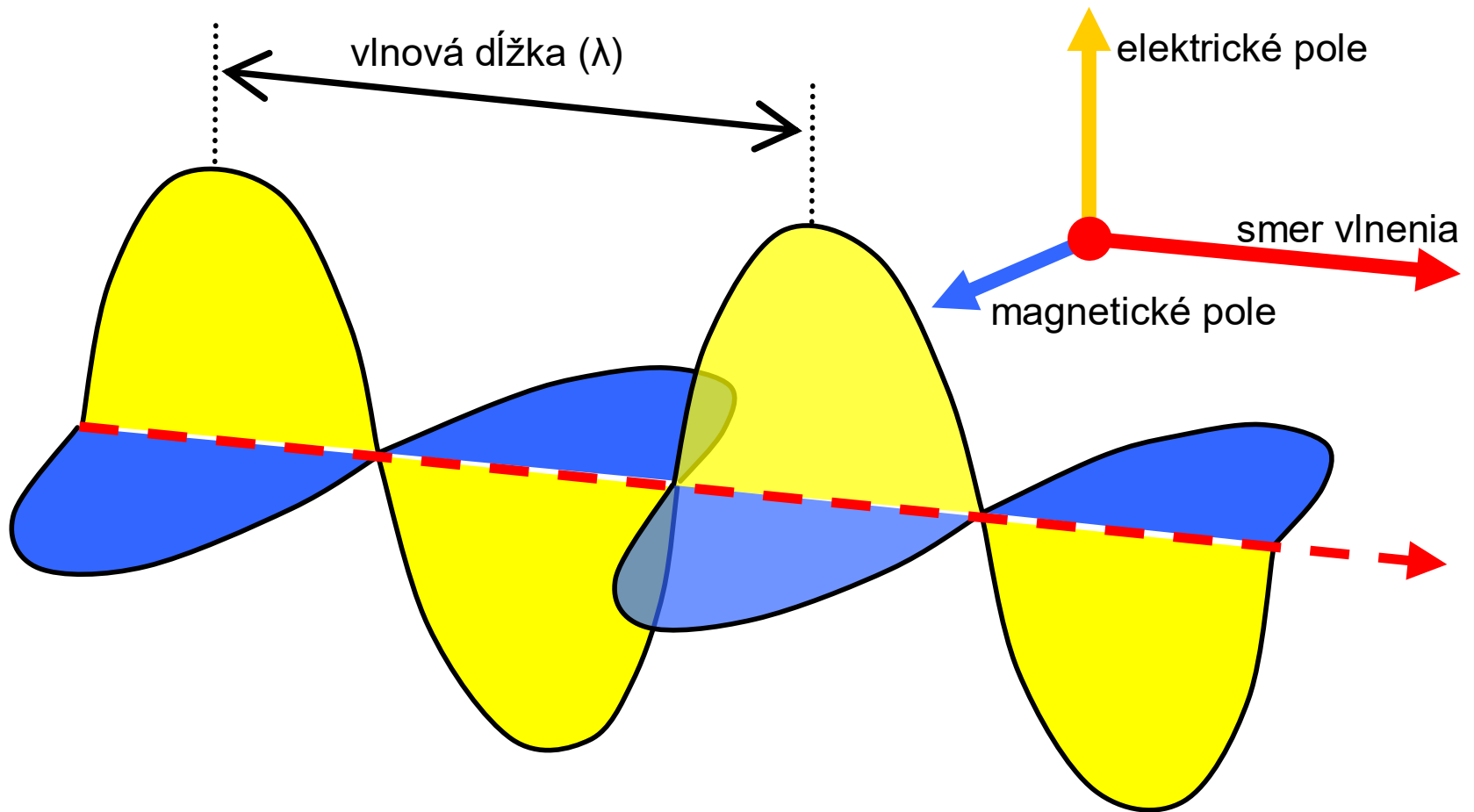
Svetelný priestor považujeme za základný pre možnosť vnímania okolitého sveta nakoľko až 80% informácií prijíma človek zrakom.

Dve reprezentácie svetla:

- **vlnová** - elektromagnetické vlnenie v oblasti 380-780 nm.
- **časticová** – prúd častíc - fotónov

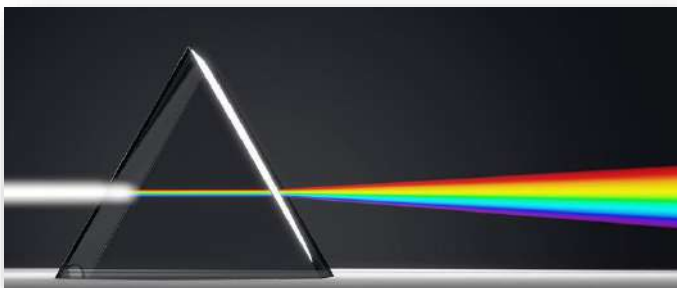
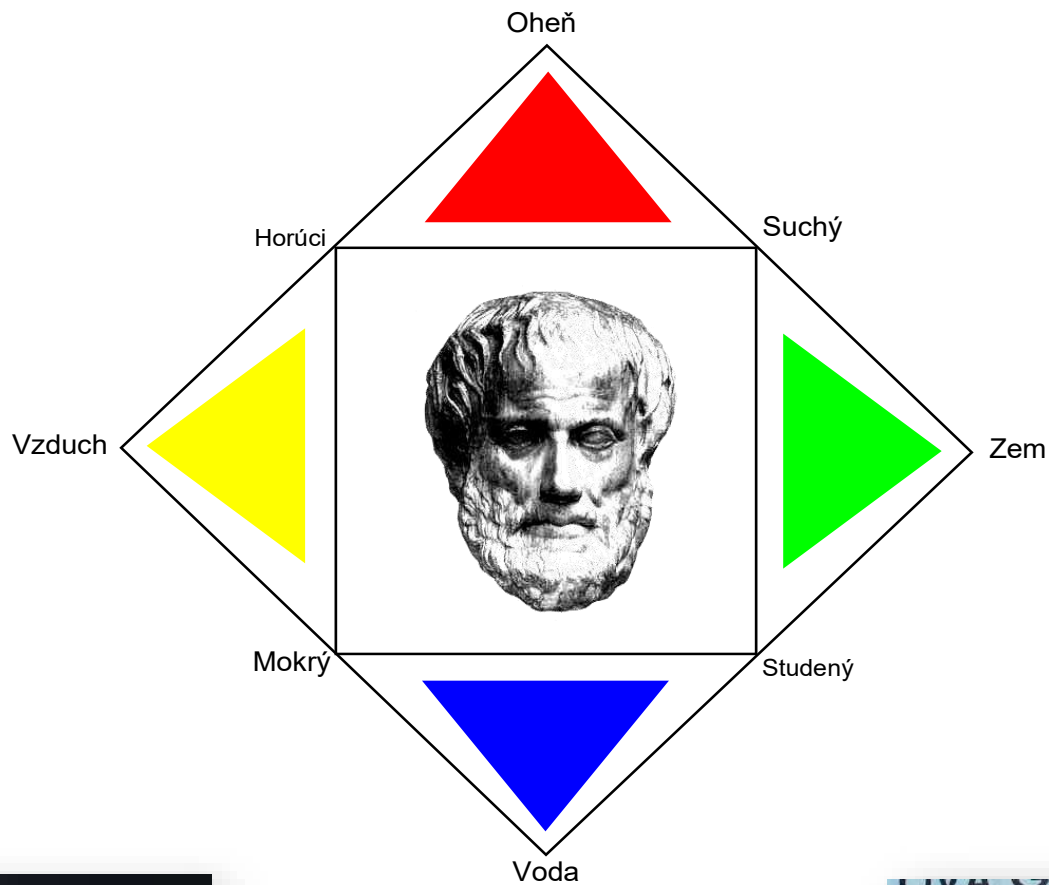


SVETLO



HISTÓRIA

Aristoteles, cca 300 p.n.l.



Optický hranol (prizma)
(Issac Newton (1665-66, publ. Opticks 1704))



FARBY

Každá farba zodpovedá určitej vlnovej dĺžke (frekvencii) vlnenia.
Rozsah farieb je:

- od **fialovej** (380nm), mimo viditeľného spektra pokračuje do ultrafialovej oblasti)
- po **červenú** (780nm), mimo viditeľného spektra pokračuje do infračervenej oblasti)



FARBY A SVETLO

Gamma lúče 0,001nm	X lúče 1nm	Ultrafialové lúče 10nm	Viditeľné svetlo 100nm	Infračervené svetlo 1000nm	Mikrovlny 1-10μm	TV rádio 100μm

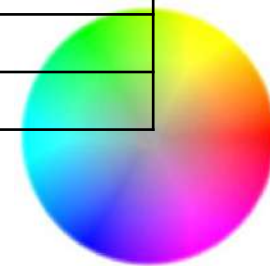


380nm

780nm

ZÁKLADNÉ FARBY A ICH VLNOVÉ DÍŽKY

Farba		Vlnová dĺžka (nm)	
aktívne farby	tmavočervená	dR	750-780
	červená	R	630-750
	oranžovočervená	OR	600-630
	oranžová	O	590-600
	žltá	Y	570-590
pasívne farby	žltozelená	YG	550-570
	zelená	G	520-550
	zelenomodrá / tyrkysová	GB/C	500-520
	modrozelená / azúrová	BG	480-500
	modrá	B	450-480
	modrofialová / indigo	BP/I	420-450
	fialová, purpurová	P/M	380-420



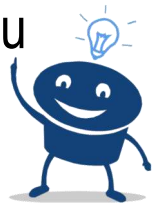
ZÁKLADNÉ TYPY A ATRIBÚTY SVETLA

Typy svetla:

- **Monochromatické** - jednozložkové
- **Achromatické** – viaczložkové

Základné atribúty svetla:

- **Farba** – je základným atribútom svetla a závisí od už spomínaj *frekvencie* (resp. vlnovej dĺžky)
- **Jas** - odpovedá vlastne *intenzite* svetla. Jasnosť zdroja svetla je v priamej úmere s intezitou (amplitúdou)
- **Sýtosť** - uvádza *čistotu svetla*. Čím vyššia je sýtosť, tým užšie je spektrum frekvencií obsiahnutých vo svetle 
- **Svetlosť** – je *veľkosť achromatickej zložky* vo svetle s určitou dominantnou frekvenciou (vlnovou dĺžkou).



FAREBNÝ PRIESTOR

Oblasť farieb pokrytá možnosťami príslušného farebného modelu sa nazýva jeho ***farebný priestor*** (*color space*)

- **Gamut** je dosiahnuteľná oblasť farieb v určitom farebnom priestore. Farby, ktoré sú mimo gamutu je možné zobrazit' v danom farebnom priestore iba približne.
- **Farebná hĺbka** je počet zobraziteľných farieb farebného priestoru a udáva sa v bitoch,
 - *farebné režimy* napr. *HighColor* (2^{16}), *TrueColor* (2^{24} , 2^{32})

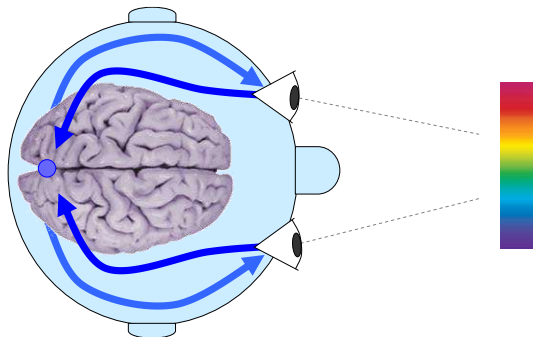


ĽUDSKÝ VIZUÁLNY VNEM

- Hlavným ľudským orgánom *prijímania* obrazových informácií je **oko**
- Hlavným ľudským orgánom *spracovania* obrazových informácií je **mozog**



To, čo človek vníma ako farbu objektu/ov, ktorý nie je zdrojom svetla, je v skutočnosti odraz referenčného (zdrojového) svetla od objektu/ov.

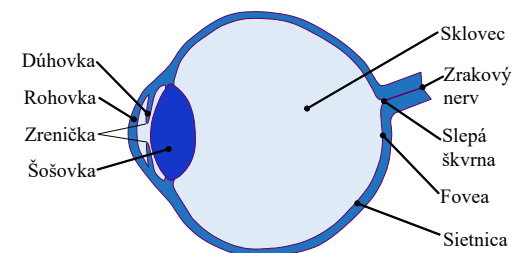


Týmto je determinované použitie technológií počítačovej grafiky

ĽUDSKÝ VIZUÁLNY VNEM

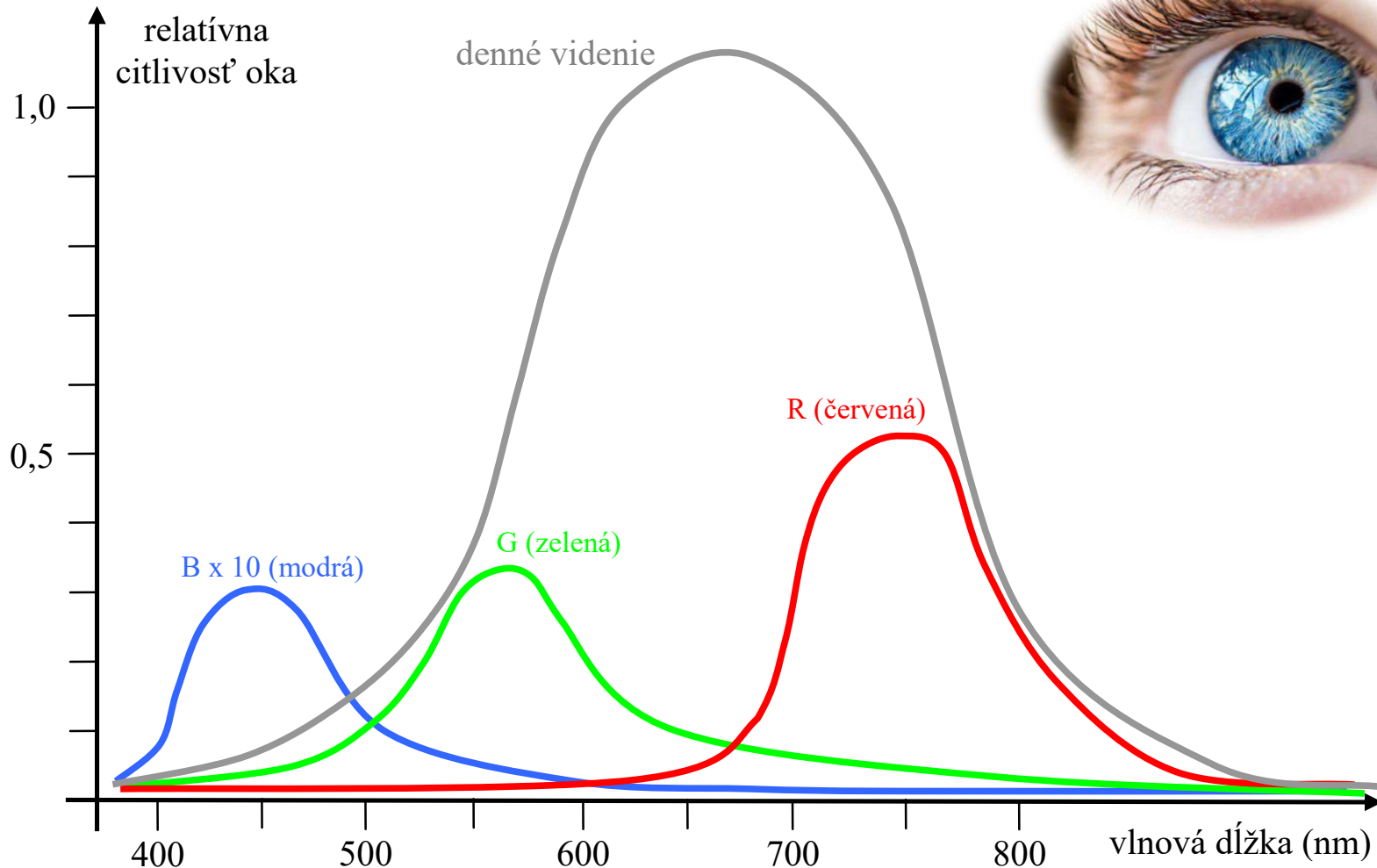
Pri prijímaní svetla sú *stimulované* (ovplyvňované) rôzne časti oka, ale najmä *receptory* oka:

- **Tyčinky** - (*tyčinkovité-rods*) počet cca 75-150 mil. rovnomerne pokrývajú celú sietnicu a umožňujú vnímať všeobecné obrazové informácie ako obrysy prípadne jas.
- **Čapíky** – (*kuželovité-cones*) počet cca 6-7 mil. sa nachádzajú v strede sietnice a sú citlivé na farby. Senzitívnosť týchto receptorov je delená do dvoch skupín, ktoré rozlišujú farebný rozdiel spektra:
 - červená-zelená (RG-cones)
 - modrá-žltá (BY-cones)

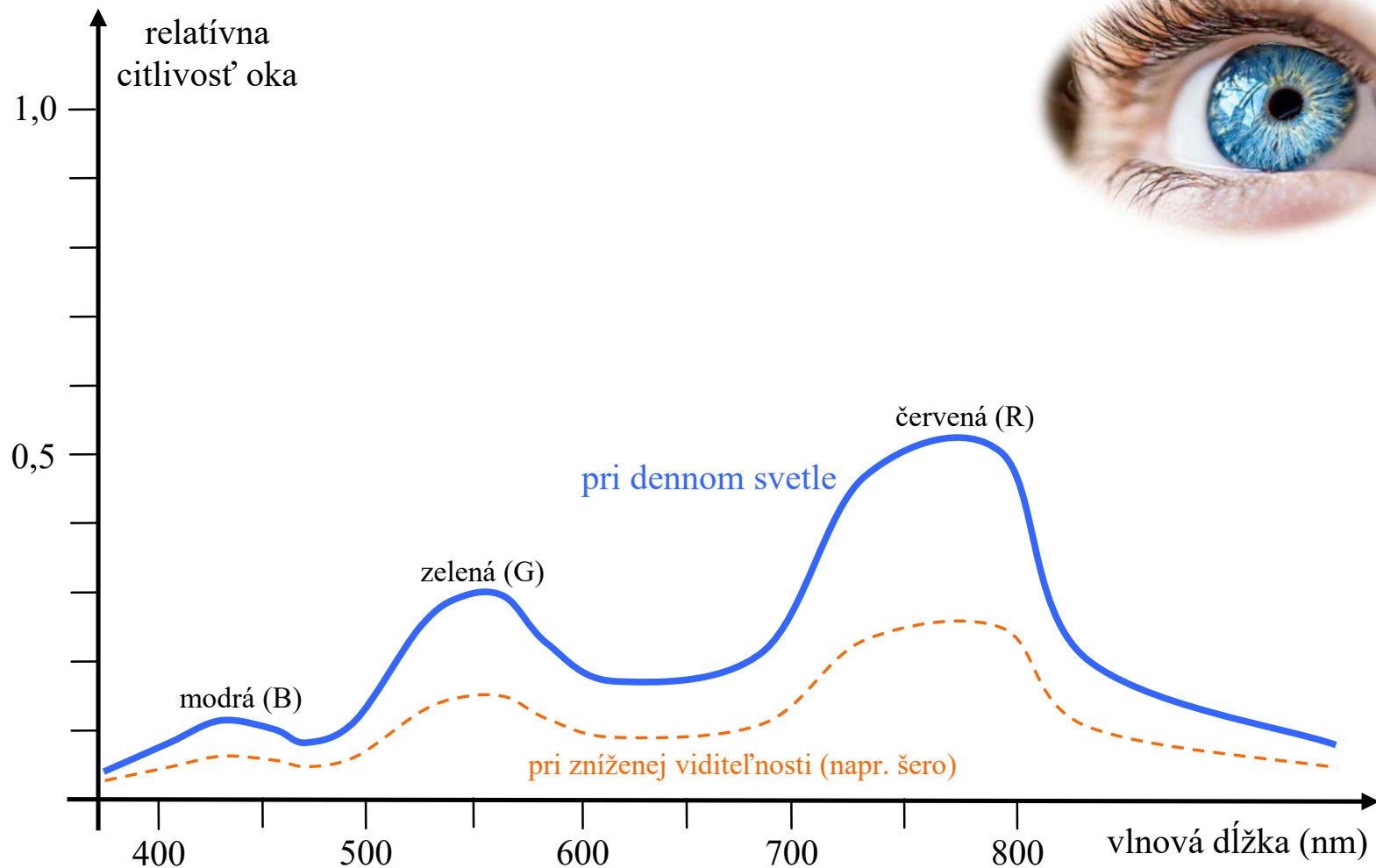


Poznámka: Rozlíšenie ľudského oka je technicky asi 140 MPx

RELATÍVNA CITLIVOSŤ ĽUDSKÉHO OKA NA FARBY (TRISTIMULUS BÁZA, SLNKO AKO REFERENČNÝ ZDROJ)

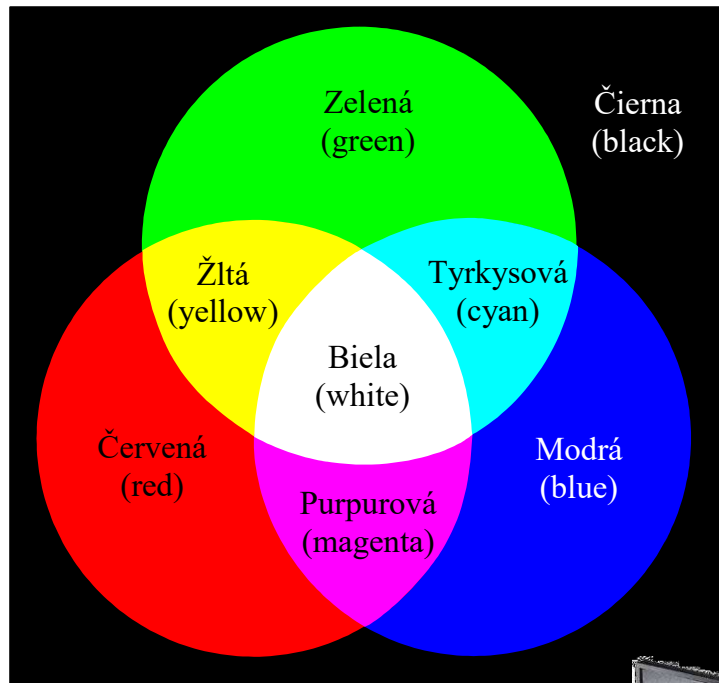


RELATÍVNA CITLIVOSŤ ĽUDSKÉHO OKA NA PRECHODY MEDZI FARBAMI (SLNKO AKO REFERENČNÝ ZDROJ)

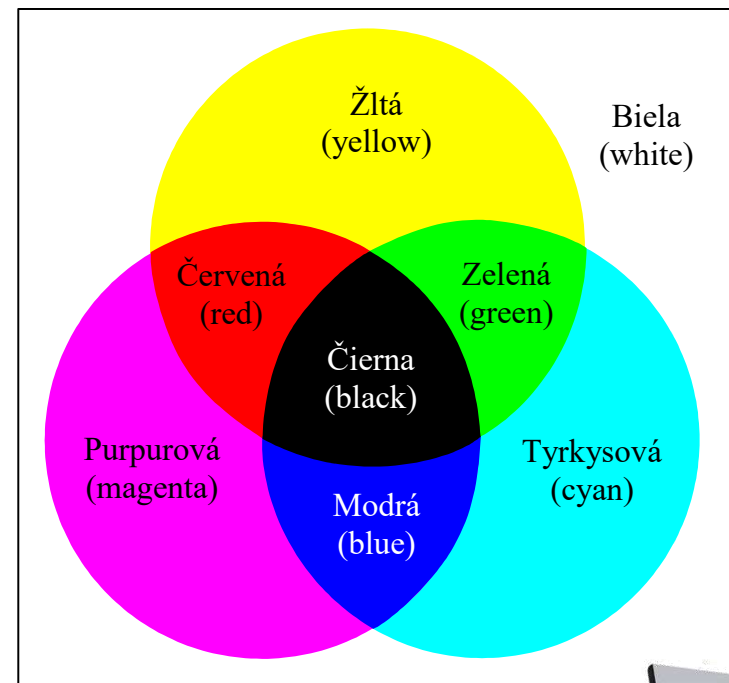


MIEŠANIE FARIEB

- Aditívne miešanie

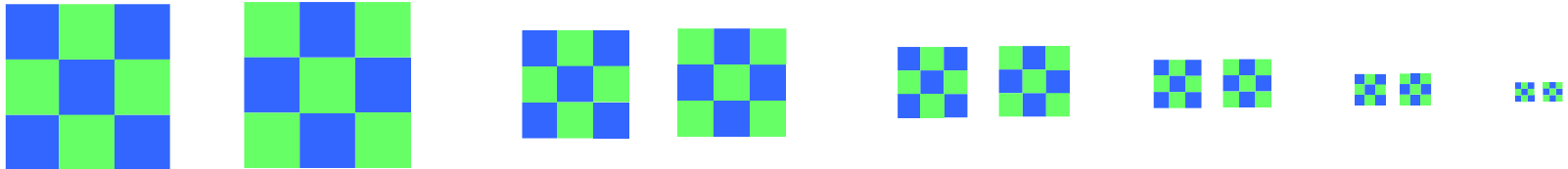


- Subtraktívne miešanie

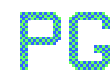
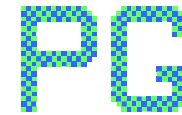
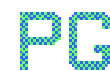
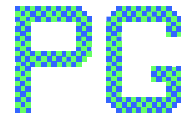
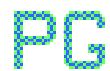
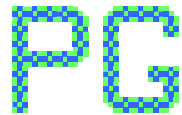
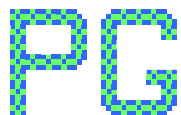
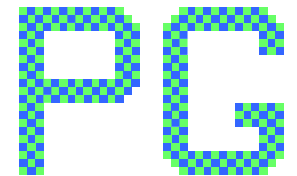
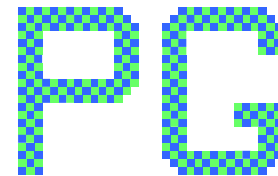
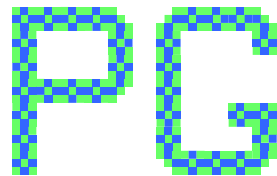
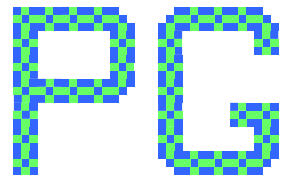
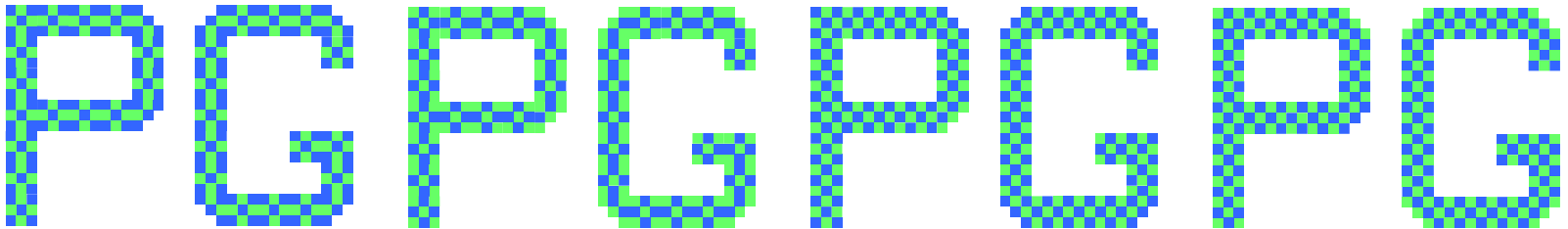


MIEŠANIE FARIEB (DITHERING)

Z POHL'ADU ČLOVEKA



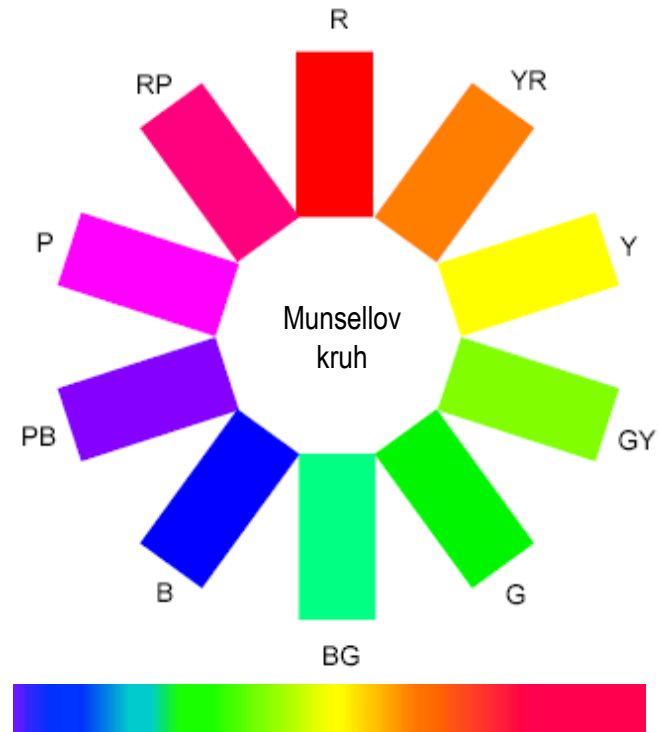
dve a viac farieb



MUNSELLOV FAREBNÝ SYSTÉM

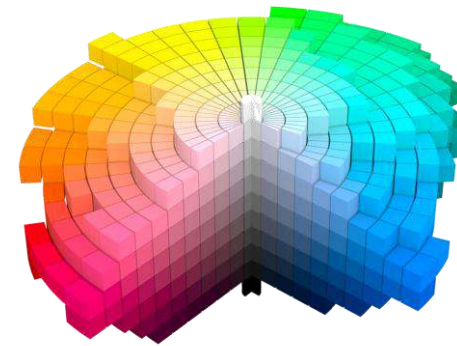


Albert Henry Munsell, 1905



Tri veci sú potrebné na to, aby človek videl farby:

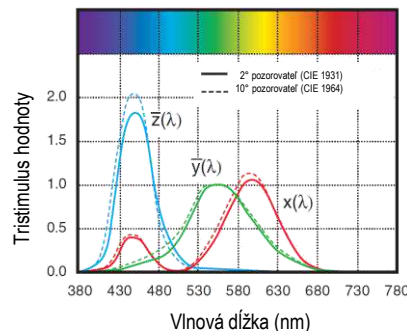
- *svetelný zdroj* (osvetlenie, illuminant)
- *objekt* (vzorka, sample)
- *pozorovateľ/spracovateľ* (observer/processor)



3D reprezentácia Munsellovho farebného systému (1943)

CIE CHROMATICKÝ DIAGRAM (1931, 1964, 1976)

- farebný model vyvinutý na základe štandardného pozorovateľa, ktorého farebné videnie je zástupcom ľudskej populácie s normálnym farebným videním (*CIE - Commission internationale de l'éclairage (fr.) - International Commission on Illumination (en)*).
- pre určenie farby sa používa trojzložkový (**tristimulus**) systém. Farba je určená trojicou čísel/súradníc (X, Y, Z).
 - Tieto trojzložkové hodnoty udávajú množstvo každej z troch hypotetických primárnych zložiek, ktoré definujú farbu. Hodnota Y dáva jas objektu a primárne zložky sú vybrané tak, aby boli viditeľné farby definované ako pozitívne hodnoty.

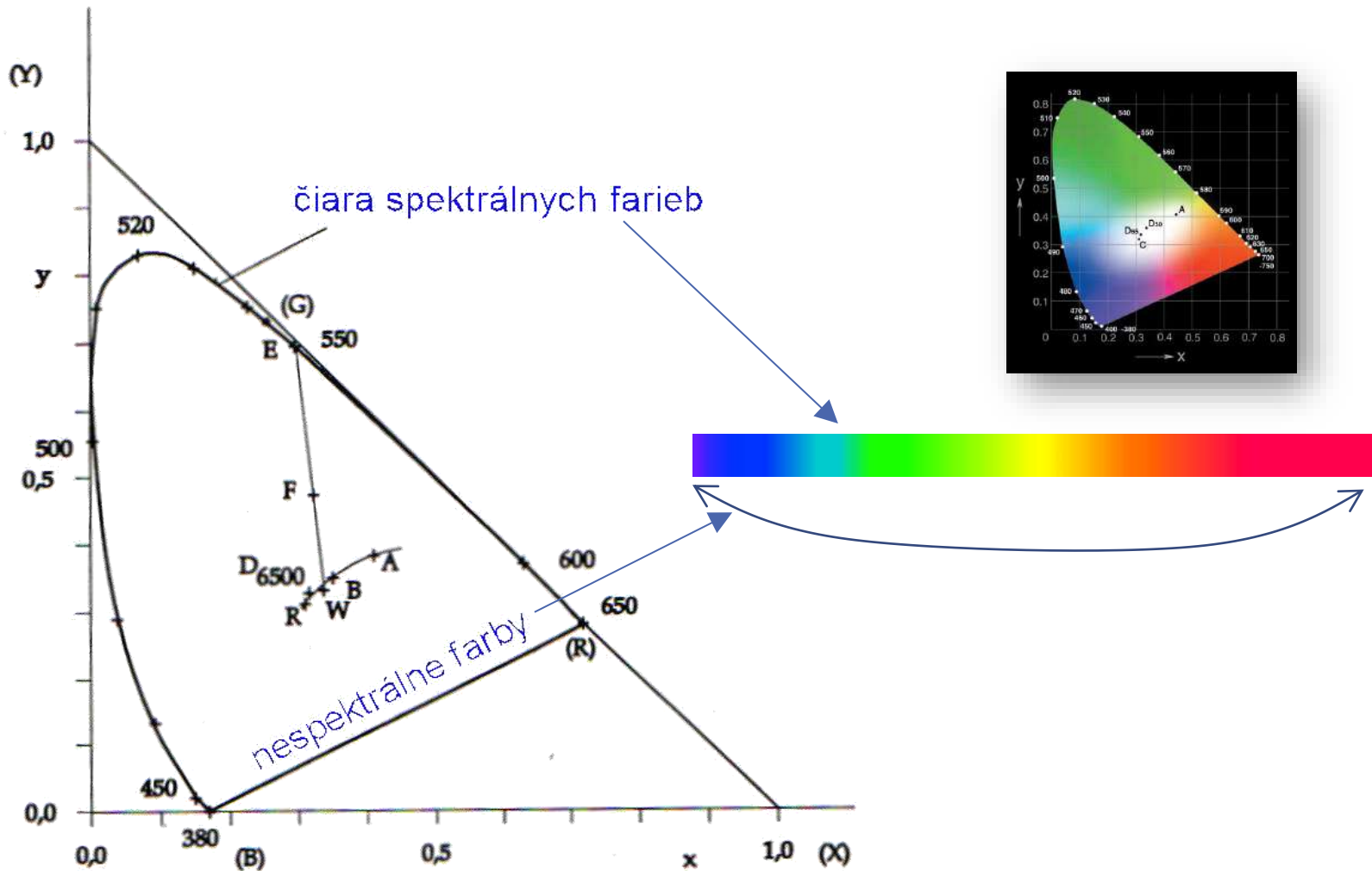


Tristimulus koncept štandardného pozorovateľa založený na priemernej ľudskej reakcii na vlnové dĺžky svetla (štandardný pozorovateľ predstavuje to, ako vidí priemerný človek farby vo viditeľnom spektre.)

Poznámka: Asi 1/10 žien má v očiach štyri druhy čapíkov (*tetrachromatia*) namiesto štandardných troch (preto tieto ženy môžu vidieť až približne 100-krát väčšiu škálu farebných odtieňov).

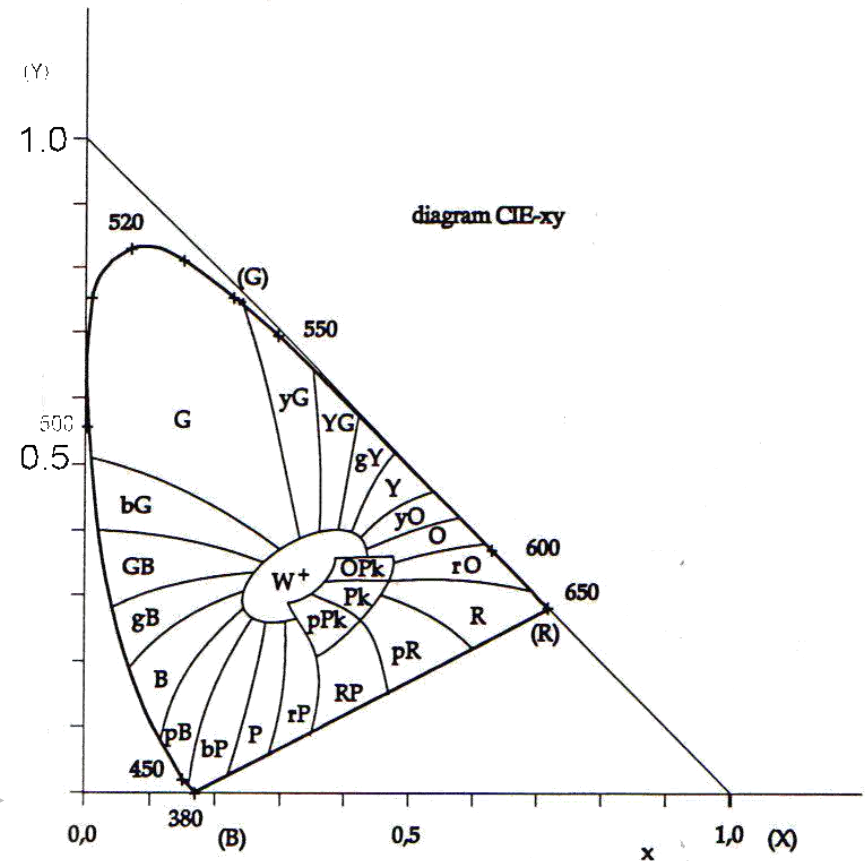
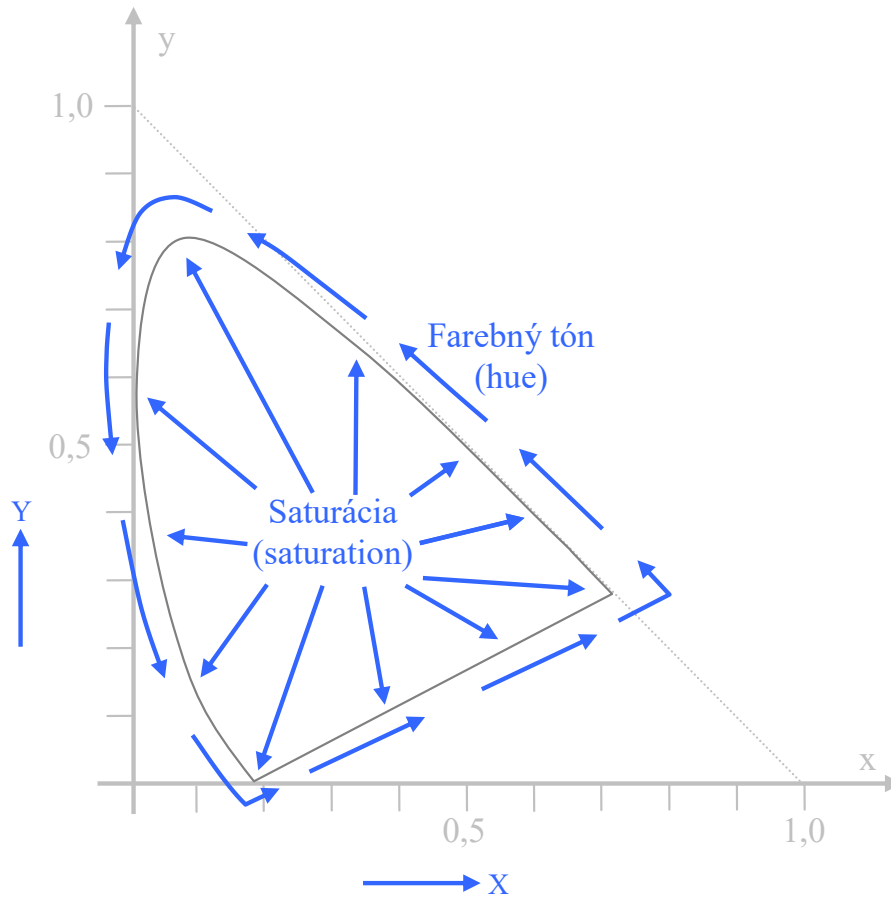
CIEXY CHROMATICKÝ DIAGRAM (1931)

ZÁKLADNÁ KONŠTRUKCIA



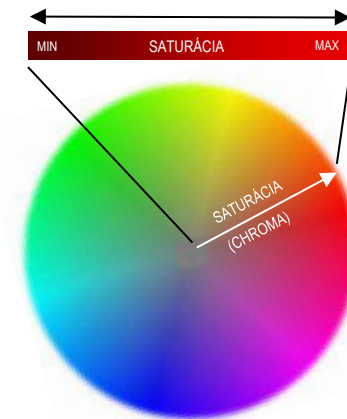
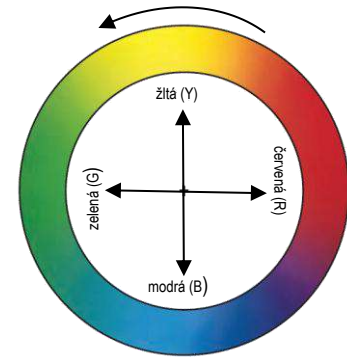
CIEXY CHROMATICKÝ DIAGRAM (1931)

ZÁKLADNÁ KONŠTRUKCIA



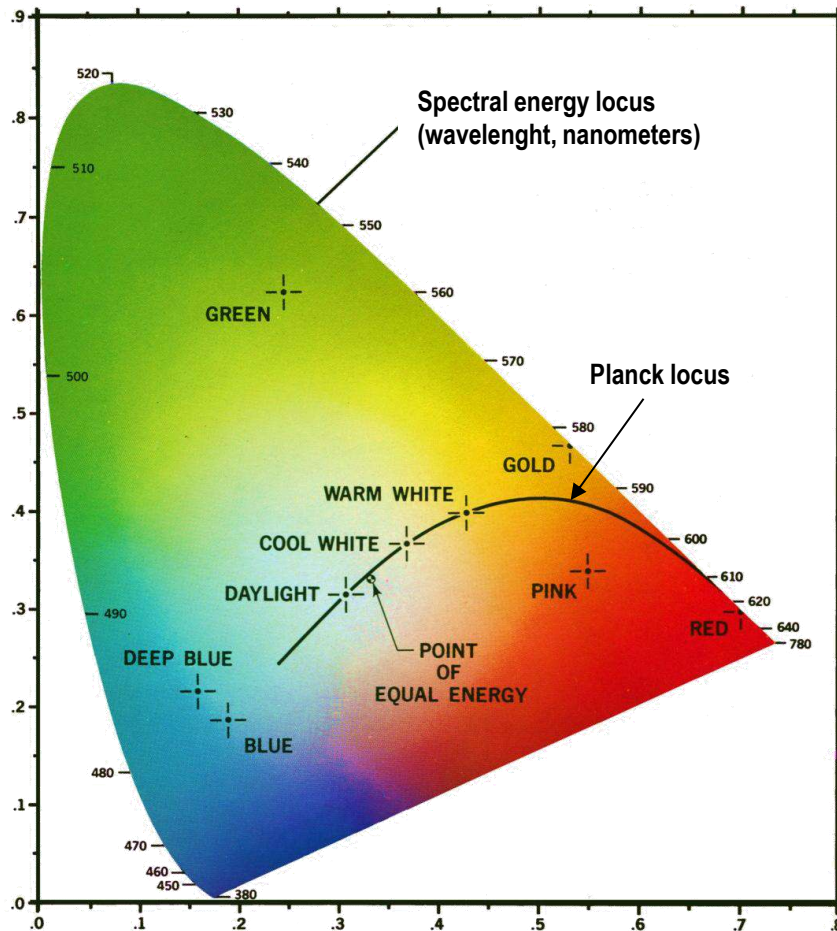
ODTIEŇ A SATURÁCIA

- *Odtieň* – (farebný tón, *Hue*) je spôsob, akým vnímame farbu objektu - červená, oranžová, zelená, modrá atď'. Definuje spôsob popisu prechodu medzi jednotlivými farbami. Najčastejšie sa používa uzavretý kruhový cyklický spôsob.
- *Saturácia* – (chroma, chromacity) - popisuje živosť alebo nevýraznosť farby t.j. ako blízko je farba k sivej alebo čistému odtieňu. Farby v strede sú sivé (matné) a sýtejšie (živšie), keď sa pohybujú smerom k obvodu.



CIE - CHROMATICKÝ DIAGRAM (1931, 1964, 1976)

UNIFORM CHROMATICITY SCALE DIAGRAM



Planckov lokus alebo **lokus čierneho telesa** - dráha alebo trajektória, ktorým by sa uberala farba rozžeraveného čierneho telesa v určitom farebnom priestore pri zmene teploty čierneho telesa. Prechádza od sýtočervenej pri nízkych teplotách cez oranžovú, žltkastobielu, bielu a nakoniec modrobielu pri veľmi vysokých teplotách.

Teplota farieb



Príklady farebných teplôt rôznych svetelných zdrojov (napr. mesiac a zrkadlo nie sú zdroje svetla):

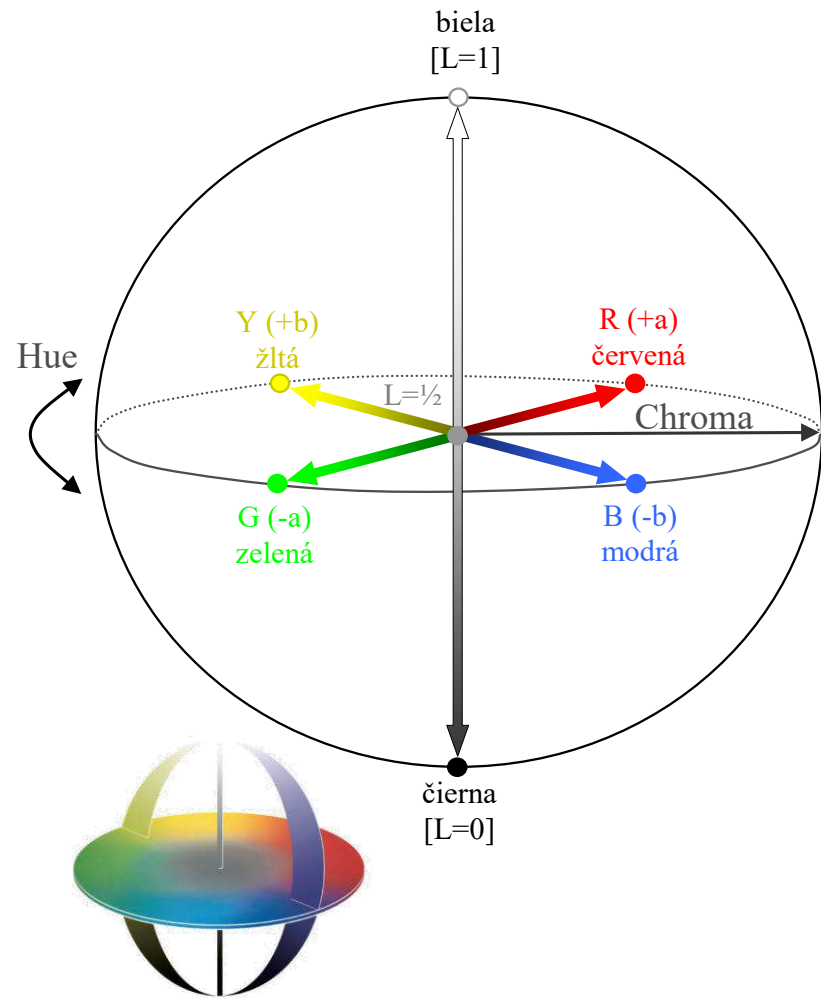
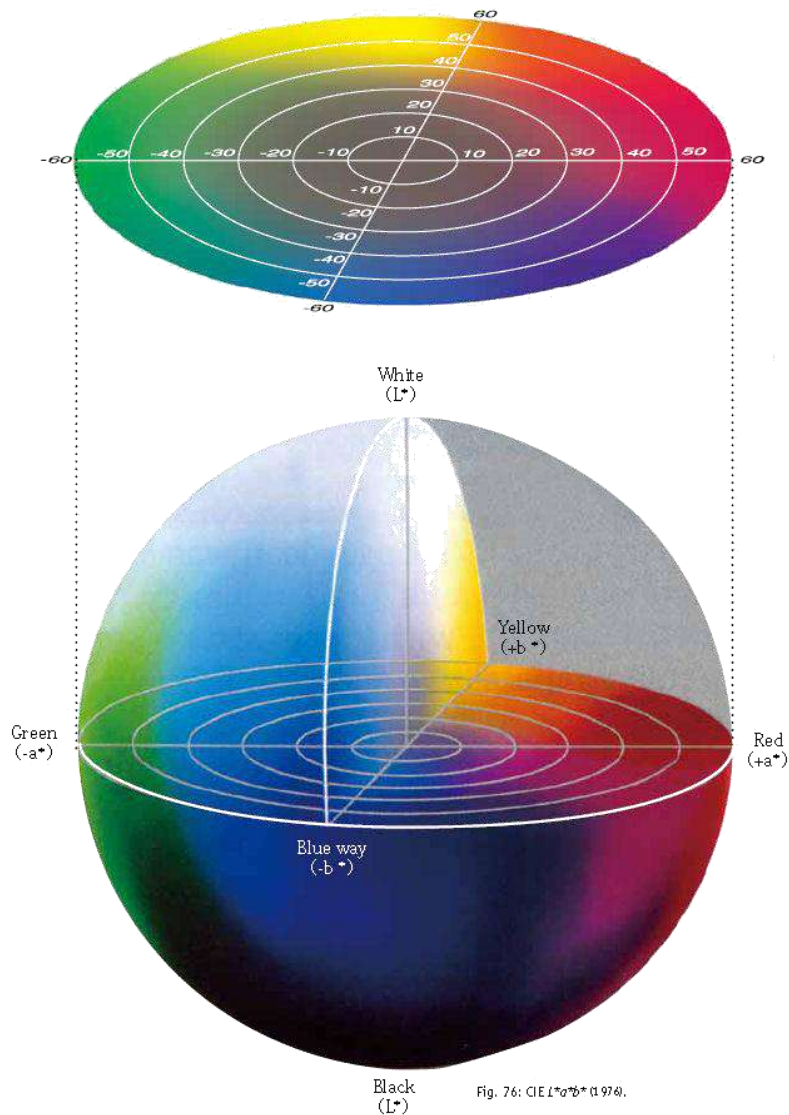
- 1200°K : sviečka
- 2800°K : žiarovka
- 3000°K : štúdiové osvetlenie
- 5000°K : denné svetlo, žiarivka
- 5500°K : fotografické blesky
- 6000°K : jasné poludňajšie svetlo
- 7000°K : mierne zamračená obloha
- 8000°K : oblačno, hmlisto
- 10 000°K : veľmi zamračená obloha

CIEXYZ CHROMATICKÝ DIAGRAM (1964, 1976)

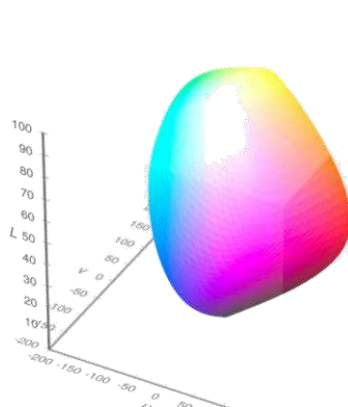
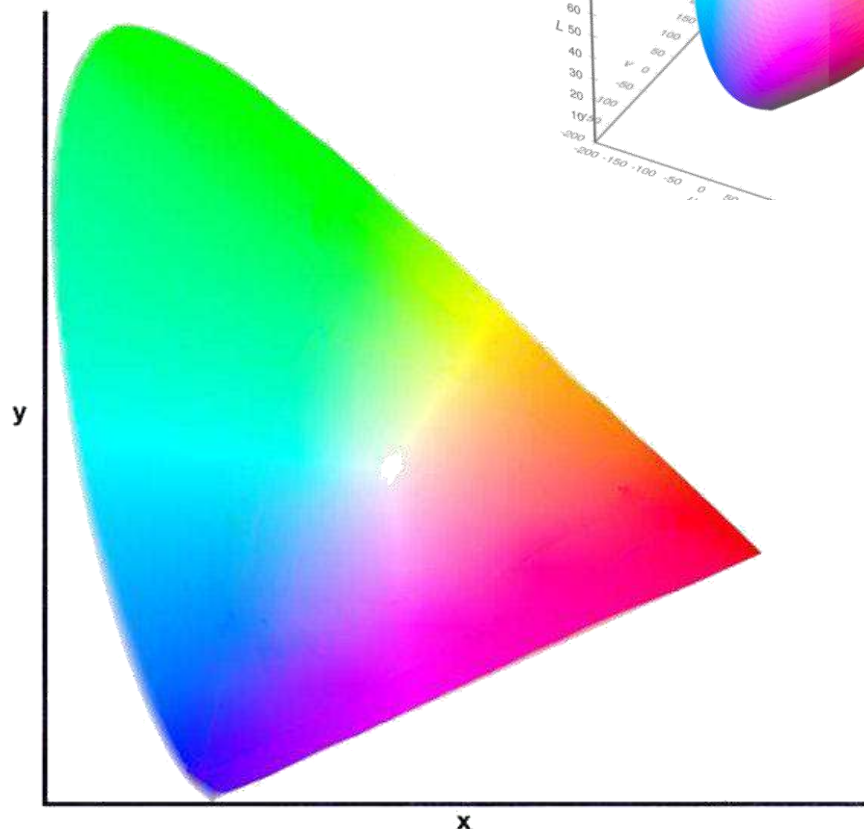
- **CIELAB** – (*svetlo absorbované*; atramenty, farbivá, pigmenty) reprezentuje popis farebného priestoru založený na subtraktívnom miešaní.
 - CIE $L^*a^*b^*$ (L- lightness, a = red-green, b=blue-yellow)
- **CIELUV** – (*svetlo emitované*; fosfor, farebné svetelné zdroje) reprezentuje popis farebného priestoru založený na *aditívnom* miešaní
 - CIE $L^*C^*h^\circ$ (L-lightness, C = chroma (saturácia), h = hue (farebný tón))

Škála svetlosti (jasnosti) je pre oba modely rovnaká a je založená na tretej odmocnine svietivosti, ktorý dáva lineárna stupnica

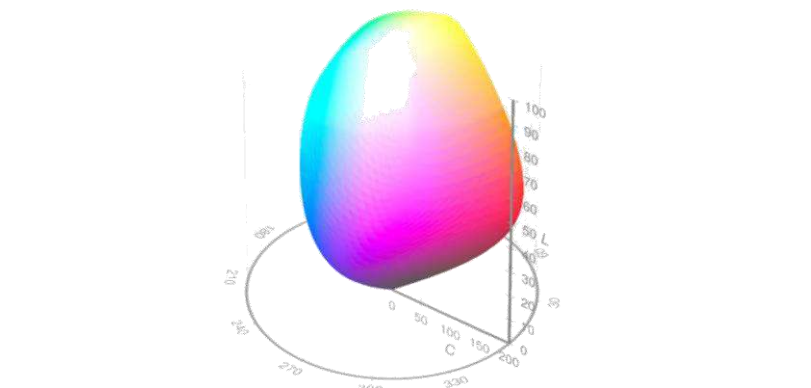
CIELAB CHROMATICKÝ DIAGRAM



CIELUV CHROMATICKÝ DIAGRAM



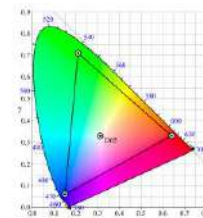
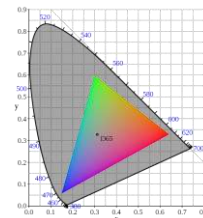
XYZ reprezentácia



Cylindrická (valcovitá) reprezentácia (CIE LCh)

ĎALŠIE FAREBNÉ PRIESTORY

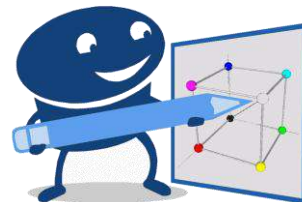
- *sRGB* je štandardný farebný priestor RGB (červená, zelená, modrá), ktorý spoločnosti HP a Microsoft vytvorili v spolupráci v roku 1996 na použitie na monitoroch, tlačiarňach a web-e (zvyčajne je to predpokladaný farebný priestor pre obrázky, ktoré nie sú označené pre iný farebný priestor ani nemajú vložený farebný profil).
- *Adobe RGB* (opRGB) je farebný priestor vyvinutý spoločnosťou Adobe Systems, Inc. (1998). Zahŕňa väčšinu farieb (50 % viditeľných farieb špecifikovaných farebným priestorom CIELAB) dosiahnuteľných na farebných tlačiarňach CMYK, ale použitím primárnych farieb RGB na zariadení, ako je napr. počítačový displej.
- *TSL* (Tint, Saturation and Lightness) je percepčný farebný priestor vyvinutý predovšetkým na účely detekcie tváre, ktorý definuje farbu ako:
 - *odtieň* (miera, do akej možno stimul opísať ako podobný alebo odlišný od iných stimulov, ktoré sú opísané ako červená, zelená, modrá, žltá, a biela, možno si predstaviť ako odtieň s pridanou bielou),
 - *sýtosť* (farebnosť stimulu v pomere k jeho vlastnému jas) a
 - *svetlosť* (jas stimulu v porovnaní so stimulom, ktorý sa v podobných podmienkach zobrazenia javí ako biely).
- *NCS* (Systém (priestor) prirodzených farieb) je patentovaný percepčný farebný priestor (model) založený na fenomenológii ľudského vnímania a nie na miešaní farieb (definuje ho farebný atlas). Je založený na troch pároch základných farieb (biela-čierna, zelená-červená a žltá-modrá) a definované farebnou opozíciou. Navrhnutý bol nemeckým fyziológom Ewaldom Heringom a udržiavaný je švédskou nadáciou Color Center Foundation.



FAREBNÉ MODELY

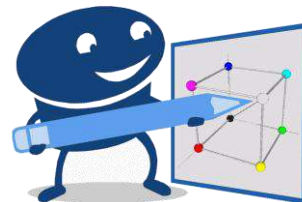
sú definované (v rámci farebnej dimenzie):

- množinou základných farieb,
- spôsobom ich miešania a
- pravidlami menenia farebných charakteristík



ZÁKLADNÉ FAREBNÉ MODELY

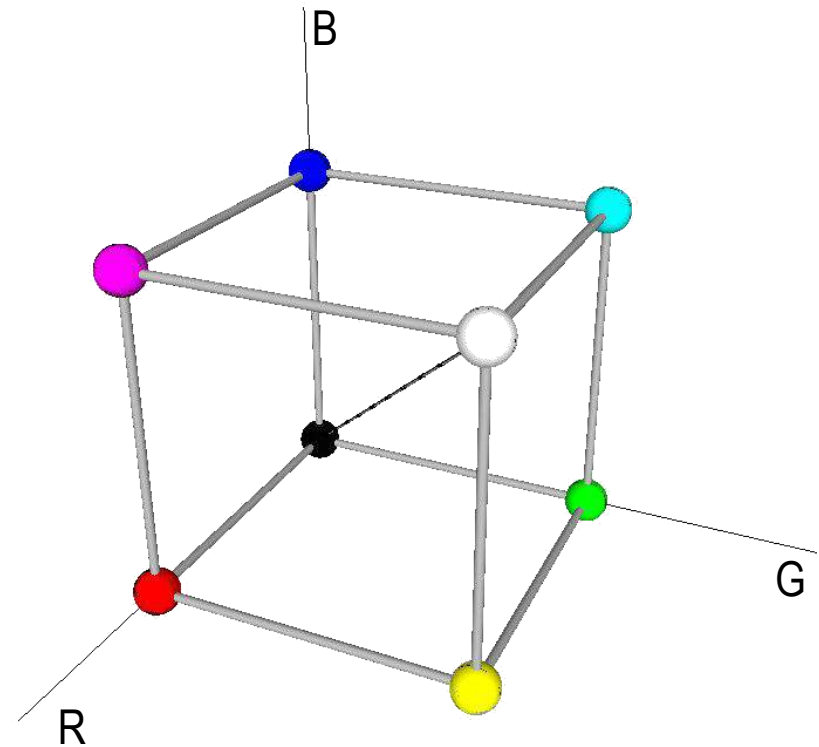
- RGB (RGBA)
- CMY (CMYK)
- HSB (HSV)
- HLS



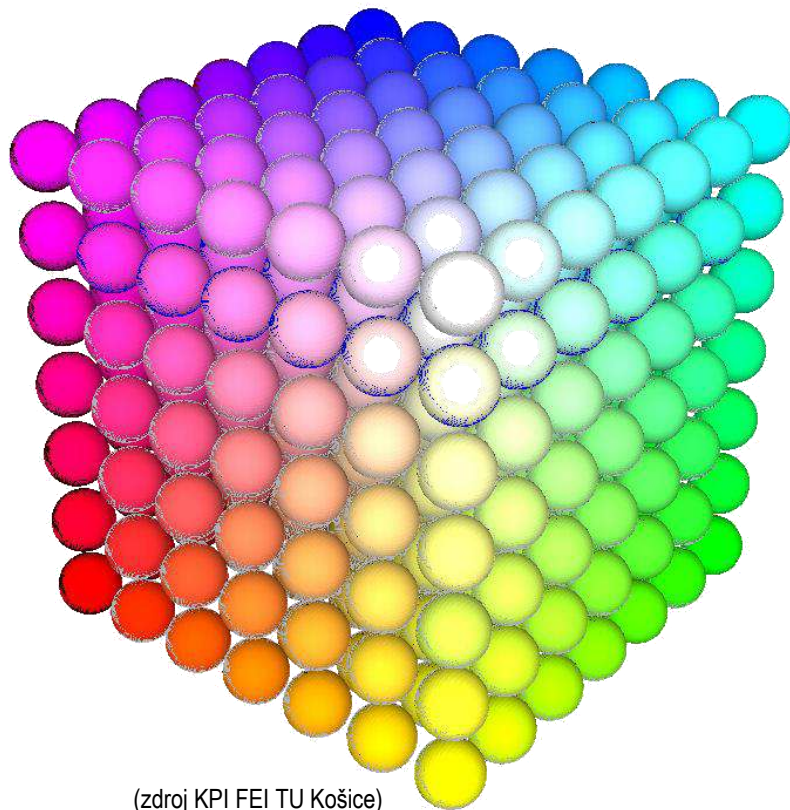
FAREBNÝ MODEL RGB

- Zložky
 - **R**ed – červená
 - **G**reen – zelená
 - **B**lue – modrá

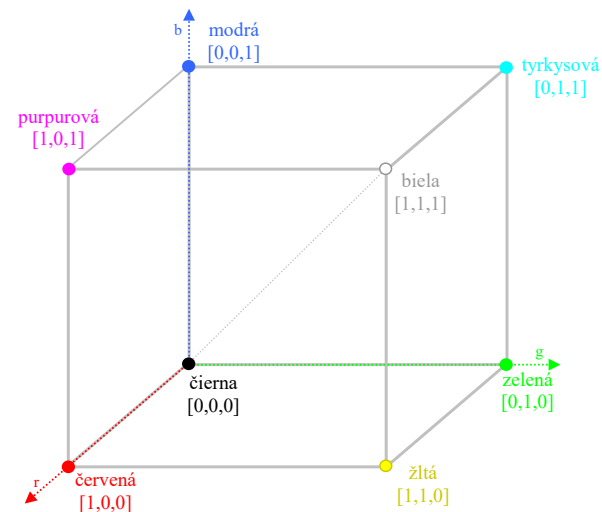
- Miešanie: aditívne
- Zmena zložky: lineárna



FAREBNÝ MODEL RGB



(zdroj KPI FEI TU Košice)



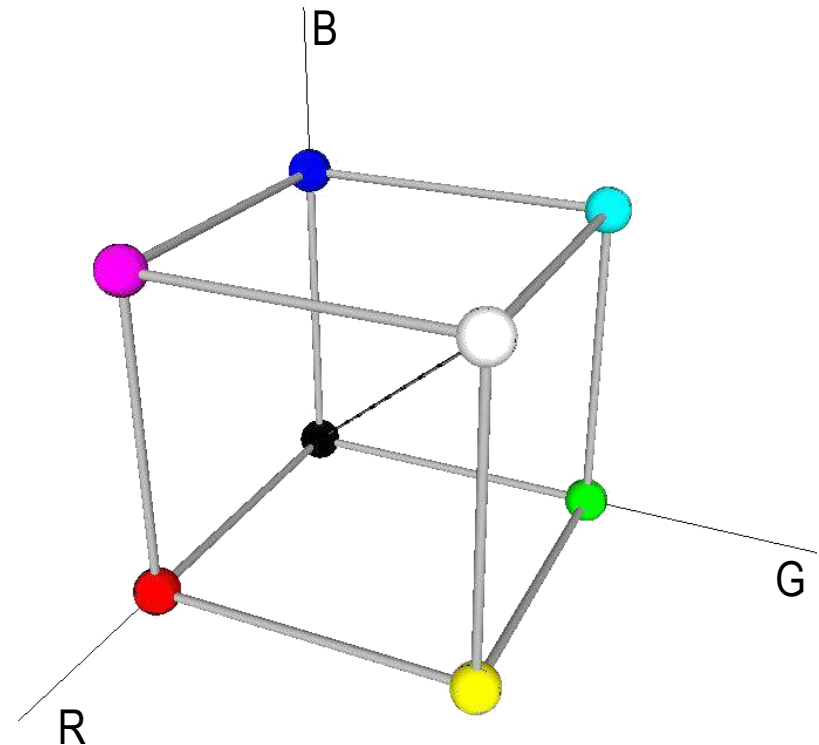
farba \ zložka	R(ed) červená	G(reen) zelená	B(lue) modrá
čierna	0	0	0
modrá	0	0	1
zelená	0	1	0
tyrkysová (azúrová)	0	1	1
červená	1	0	0
purpurová (fialová)	1	0	1
žltá	1	1	0
biela	1	1	1

FAREBNÝ MODEL RGBA

- Zložky
 - **R**ed – červená
 - **G**reen – zelená
 - **B**lue – modrá
 - **A**lfa - priesvitnosť

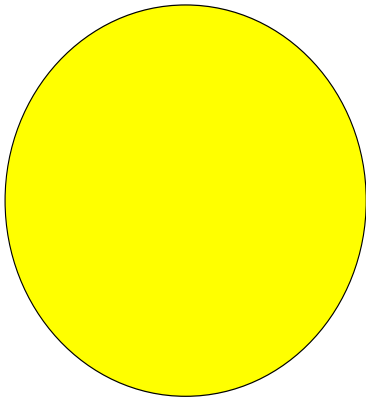
- Miešanie: aditívne

- Zmena zložky: lineárna

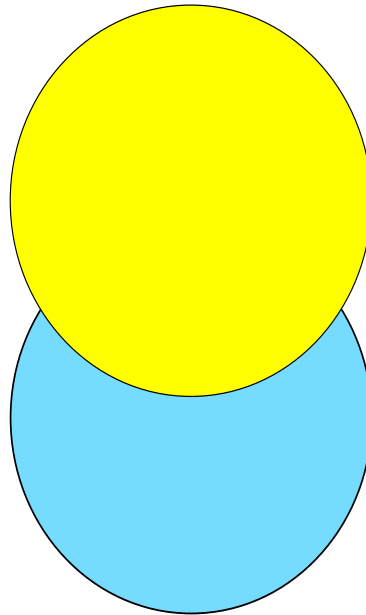


FAREBNÝ MODEL RGBA

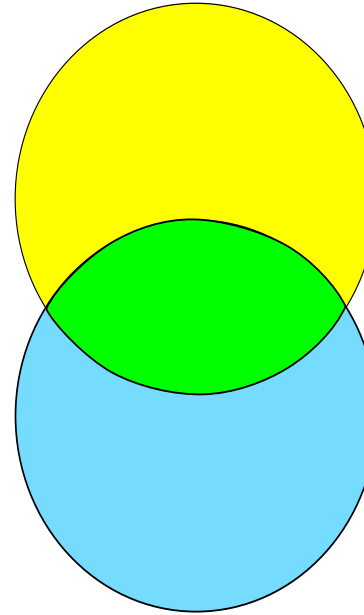
ALFA MIEŠANIE



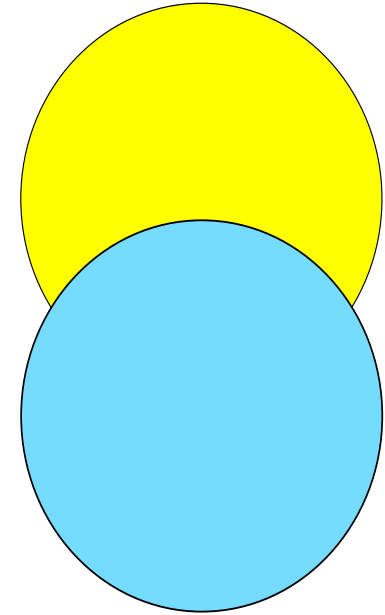
Definovanie 1. objektu



Dodanie 2. objektu
($\alpha=0$)



Dodanie 2. objektu
($\alpha=0,5$)



Dodanie 2. objektu
($\alpha=1$)

FAREBNÝ MODEL RGBA

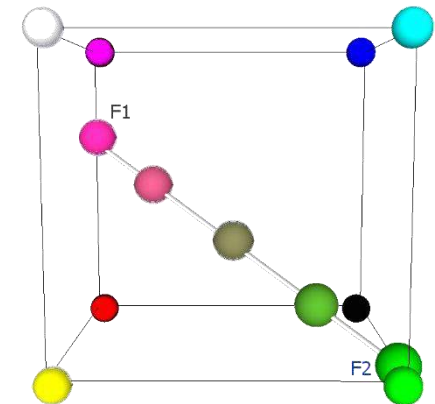
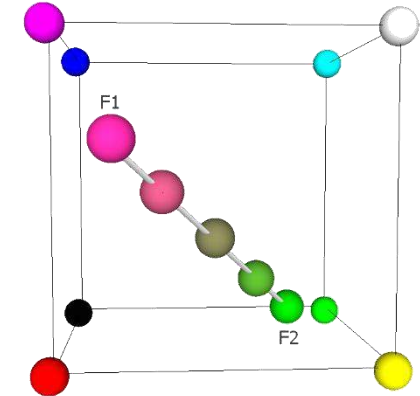
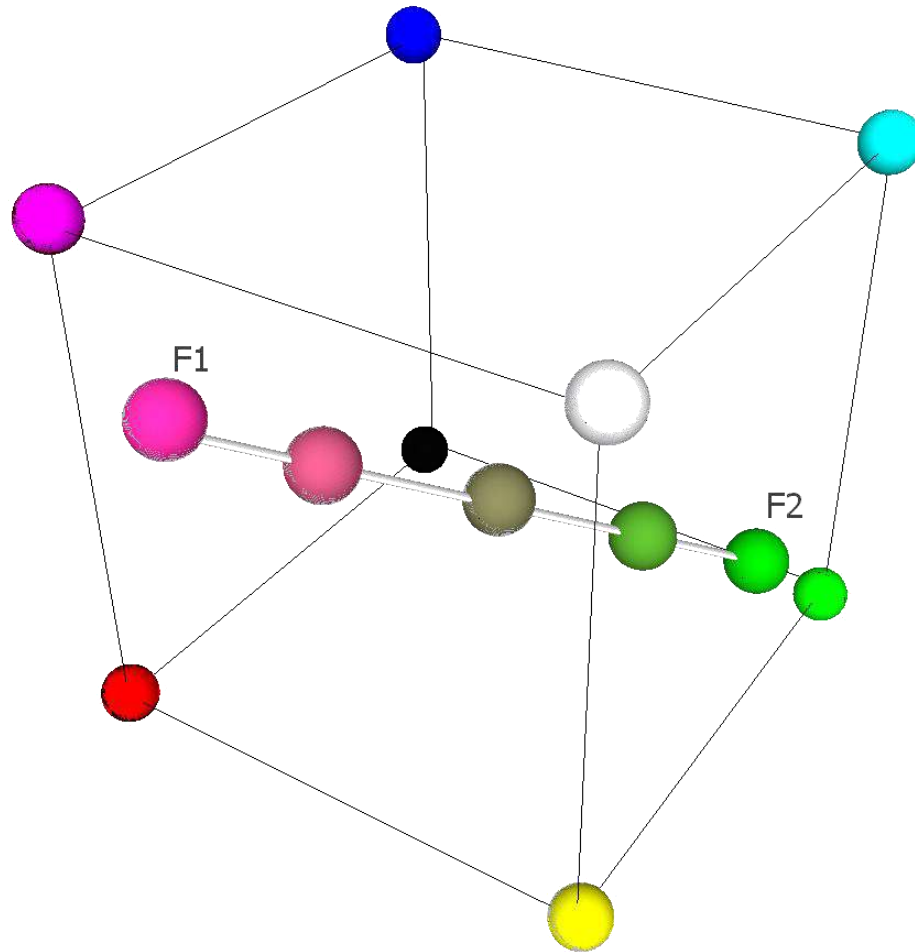
ALFA MIEŠANIE - TYPY

- Trajektória prechodu farieb
 - lineárna
 - nelineárna
- Spôsob/rozdelenie prechodu farieb
 - uniformný (pravidelný)
 - neuniformný (nepravidelný)
 - exponenciálny
 - logaritmický
 - ...



FAREBNÝ MODEL RGBA

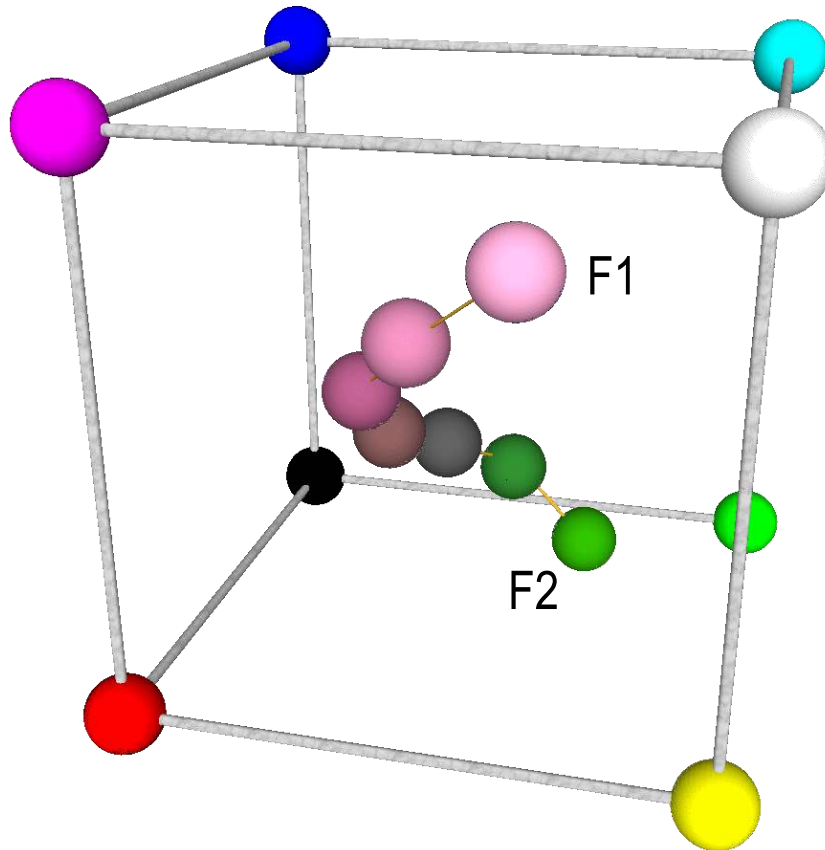
ALFA MIEŠANIE – LINEÁRNE (LINEÁRNA TRAJEKTÓRIA), RGB PRIESTOR



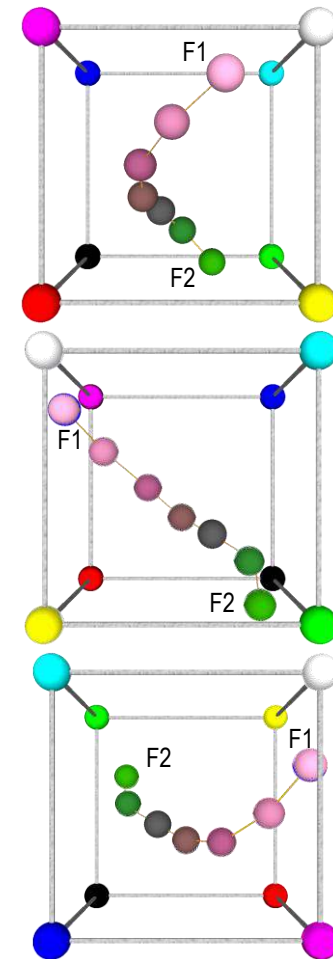
(zdroj KPI FEI TU Košice)

FAREBNÝ MODEL RGBA

ALFA MIEŠANIE - NELINEÁRNE, RGB PRIESTOR



Nelineárny prechod (NURBS trajektória)
 $F(\text{arba})_1 - F(\text{arba})_2$ pri alfa miešaní, RGB priestor

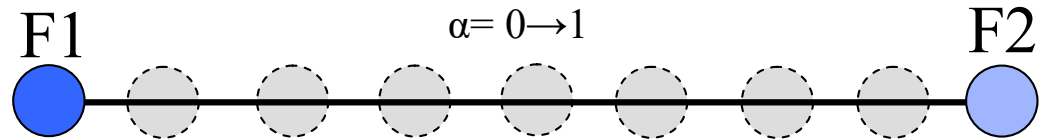


(zdroj KPI FEI TU Košice)

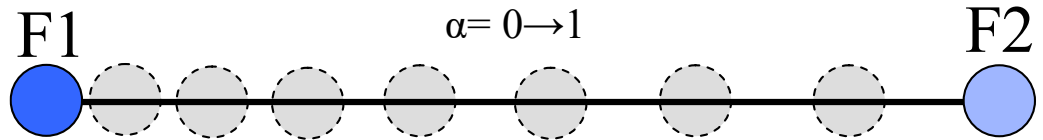
FAREBNÝ MODEL RGBA

ALFA MIEŠANIE – PRÍKLADY SPÔSOBOV PRECHODU FARIEB

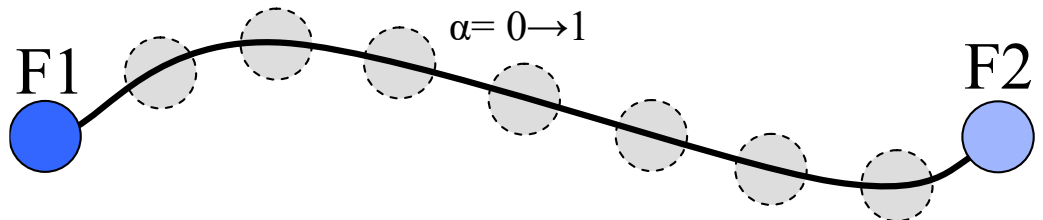
- lineárne, uniformné



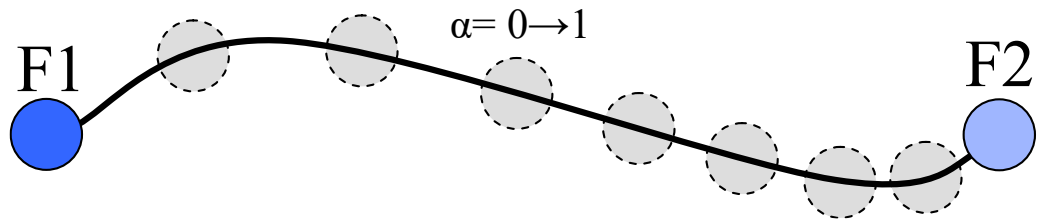
- lineárne, exponenciálne



- nelineárne, uniformné



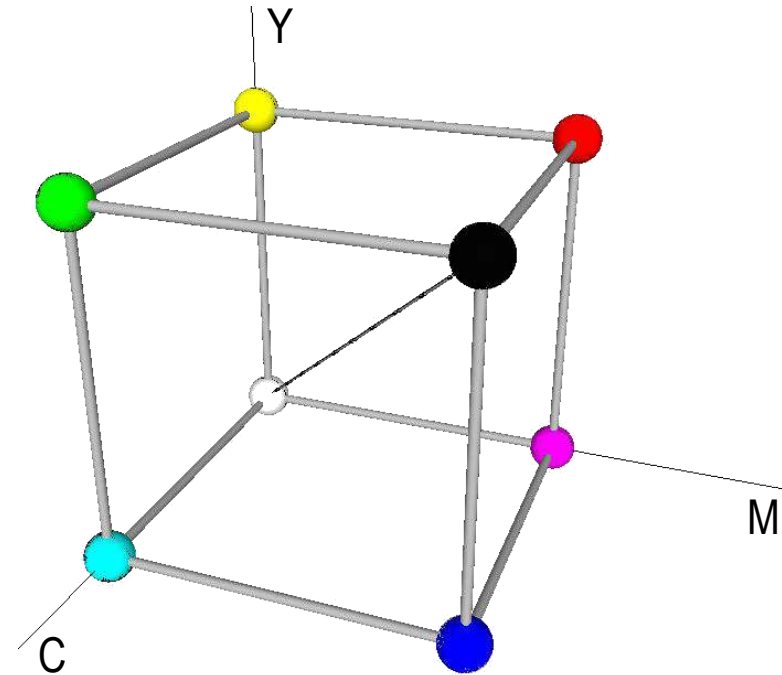
- nelineárne, logaritmické



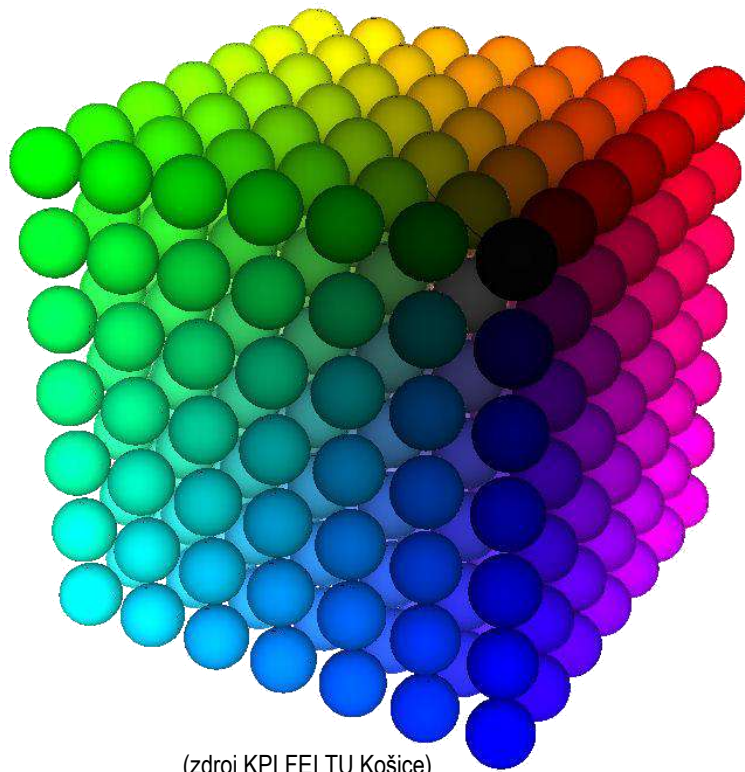
FAREBNÝ MODEL CMY

- Zložky
 - **C**yan – tyrkysová (azúrová)
 - **M**agenta – purpurová (fialová)
 - **Y**ellow – žltá

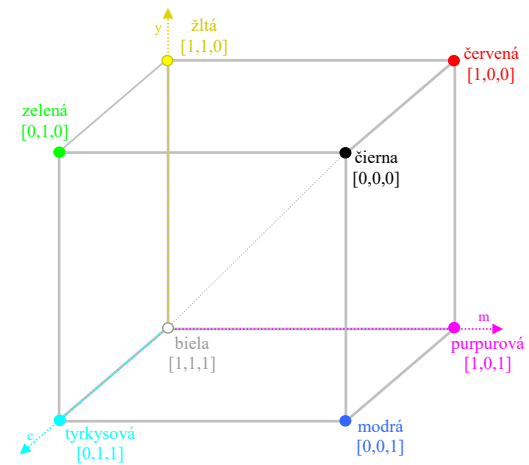
- Miešanie: subtraktívne
- Zmena zložky: lineárna



FAREBNÝ MODEL CMY



(zdroj KPI FEI TU Košice)

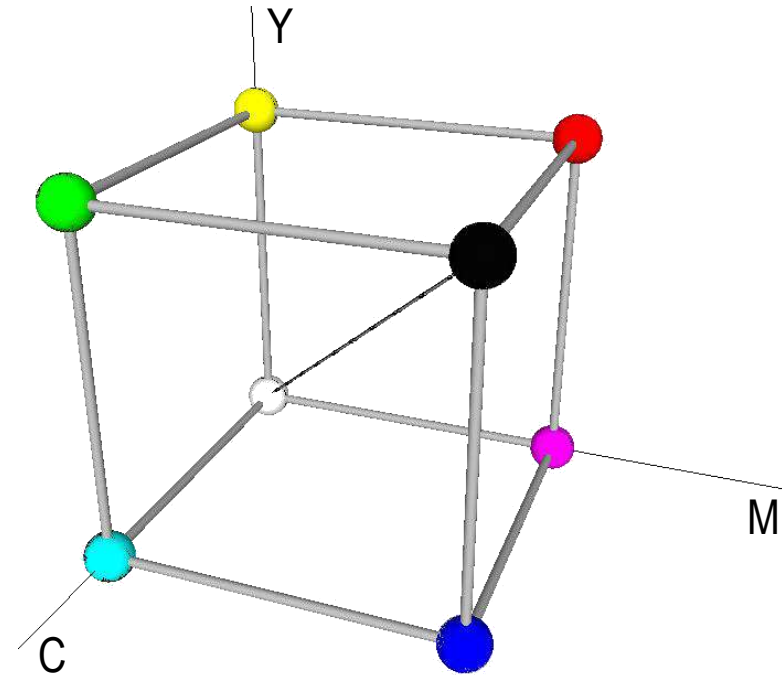


farba \ zložka	C(yan) tyrkysová	M(agenta) purpurová	Y(ellow) žltá
biela	0	0	0
žltá	0	0	1
purpurová (fialová)	0	1	0
červená	0	1	1
tyrkysová (azúrová)	1	0	0
zelená	1	0	1
modrá	1	1	0
čierna	1	1	1

FAREBNÝ MODEL CMYK

- Zložky
 - **C**yan – tyrkysová (azúrová)
 - **M**agenta – purpurová (fialová)
 - **Y**ellow – žltá
 - **blackK** - čierna

- Miešanie: subtraktívne
- Zmena zložky: lineárna



PREVOD RGB ($\langle 0-1 \rangle$) $\leftrightarrow$ CMY ($\langle 0-1 \rangle$)

RGB $\rightarrow$ CMY

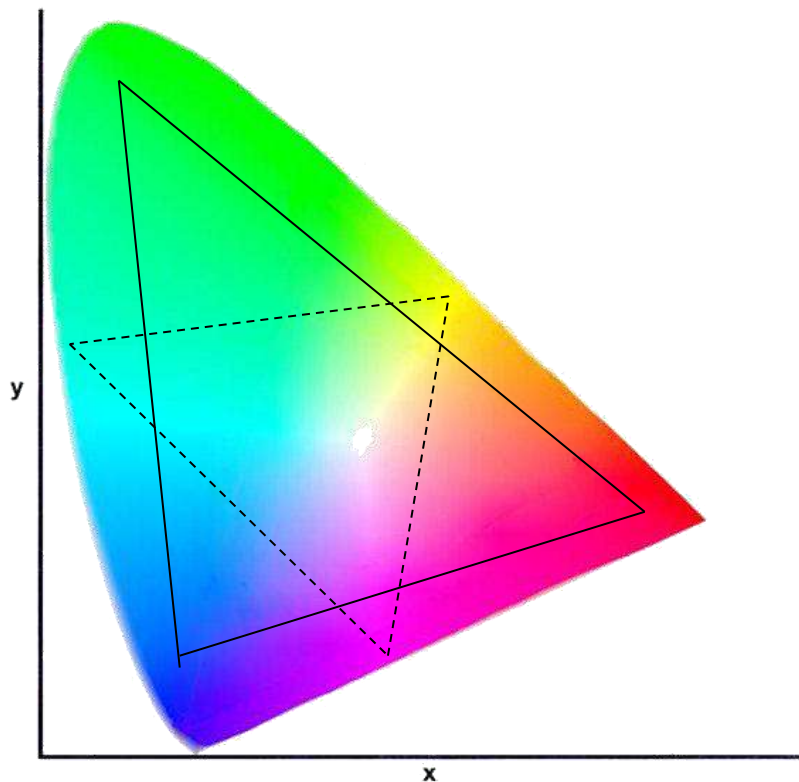
$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

CMY $\rightarrow$ RGB

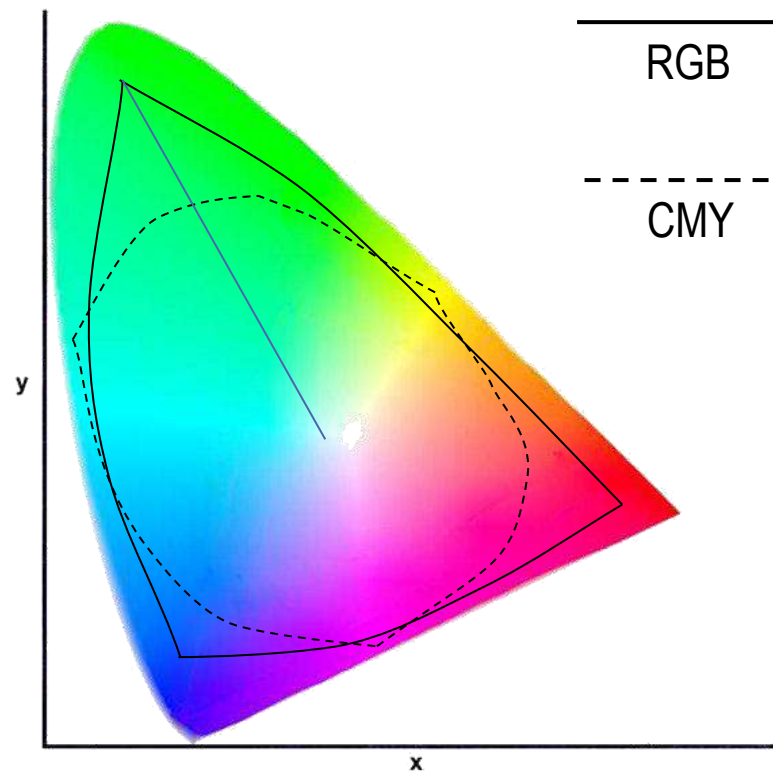
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} C \\ M \\ Y \end{bmatrix}$$



POROVNANIE RGB A CMY FAREBNÉHO PRIESTORU



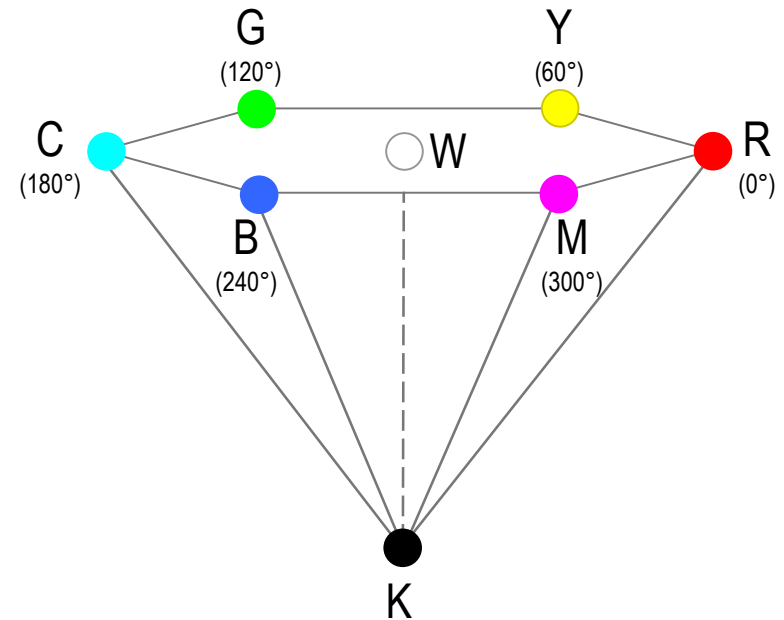
linearizované



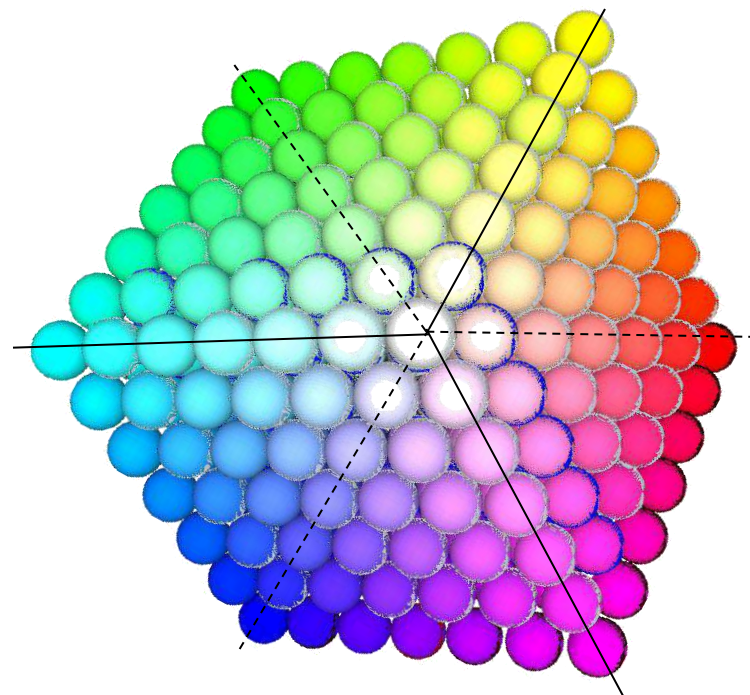
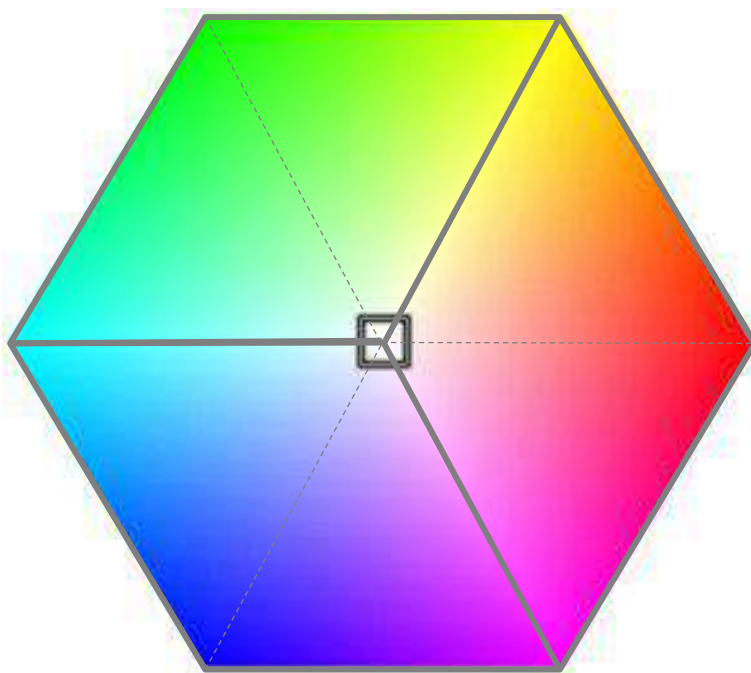
nonlinearizované

FAREBNÝ MODEL HSB (HSV)

- Zložky
 - **Hue** – farebný tón
 - **Saturation** – saturácia
 - **Brightness** – jas (value)
- Miešanie: aditívne
- Zmena zložky: uhlová a lineárna

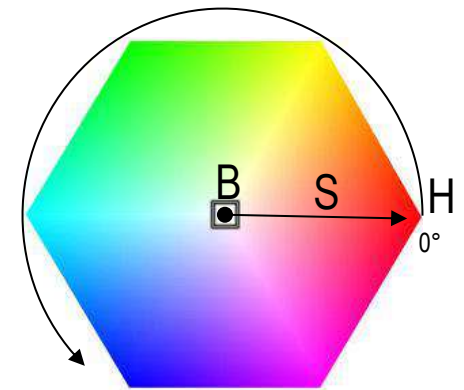
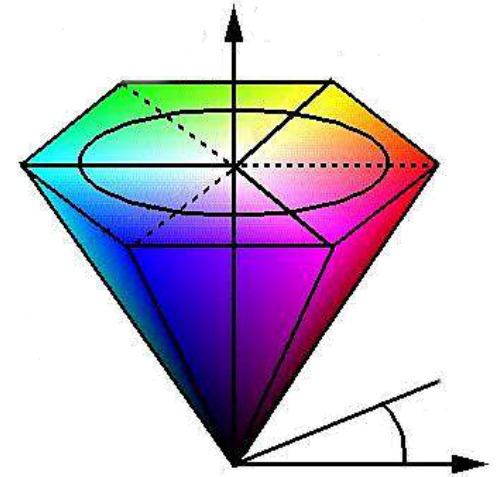
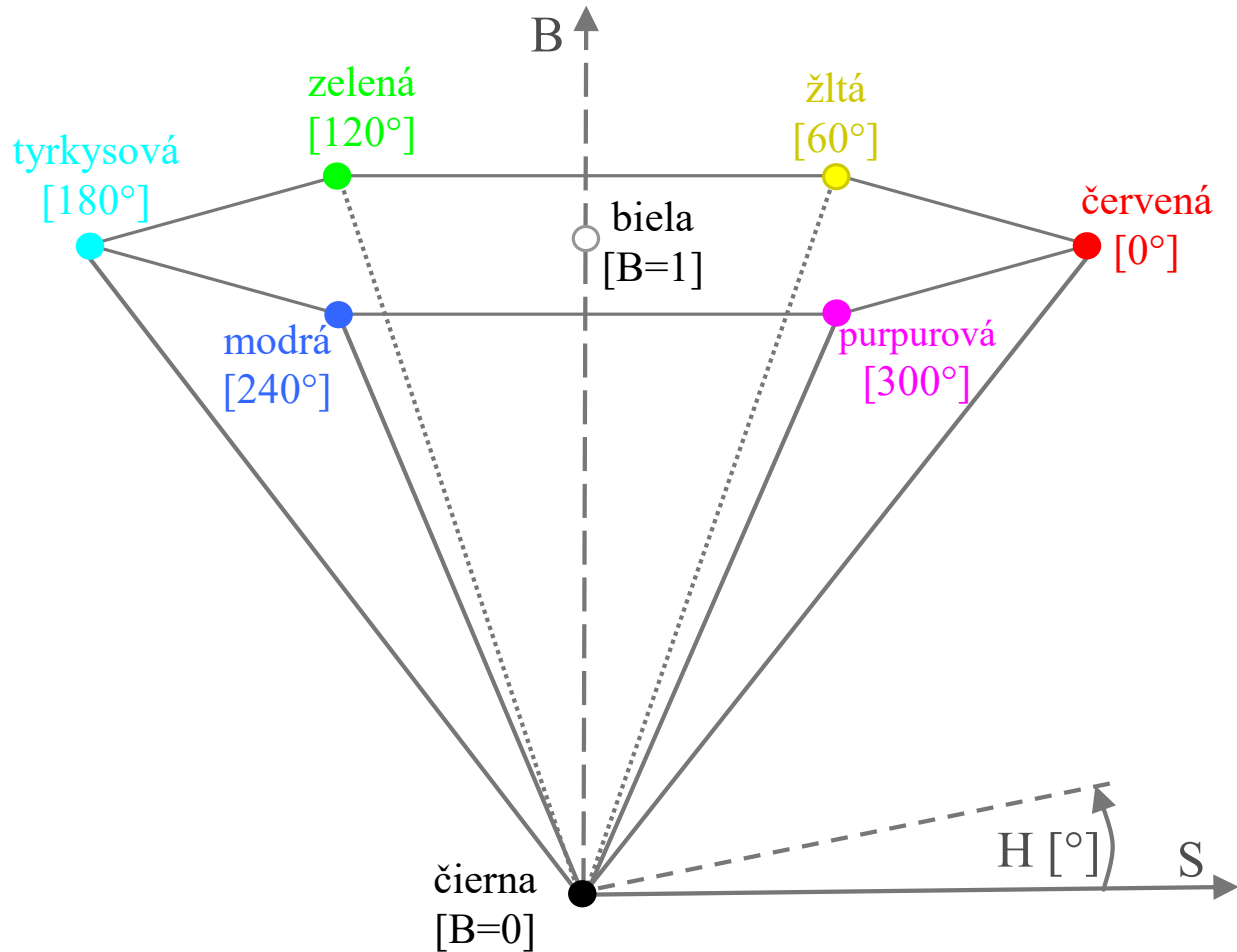


FAREBNÝ MODEL HSB (HSV)



RGB priemet do roviny kolmej k šedej diagonále $(0,0,0) - (1,1,1)$

FAREBNÝ MODEL HSB (HSV)



PREVOD RGB $\leftrightarrow$ HSB (HSV)

RGB $\rightarrow$ HSB (HSV)

1. $\max = \max(R, G, B)$
2. $\min = \min(R, G, B)$
3. $\text{delta} = \max - \min$
4. $V = \max$
5. ak $\max == 0$ $S = 0$ inak $S = (\max - \min) / \max$
6. ak prevláda zložka R: $H = (G - B) / \text{delta}$
7. ak prevláda zložka G: $H = 2 + (B - R) / \text{delta}$
8. ak prevláda zložka B: $H = 4 + (R - G) / \text{delta}$
9. $H = H * 60$, ak H je záporné číslo: $H = H + 360$

HSB (HSV) $\rightarrow$ RGB

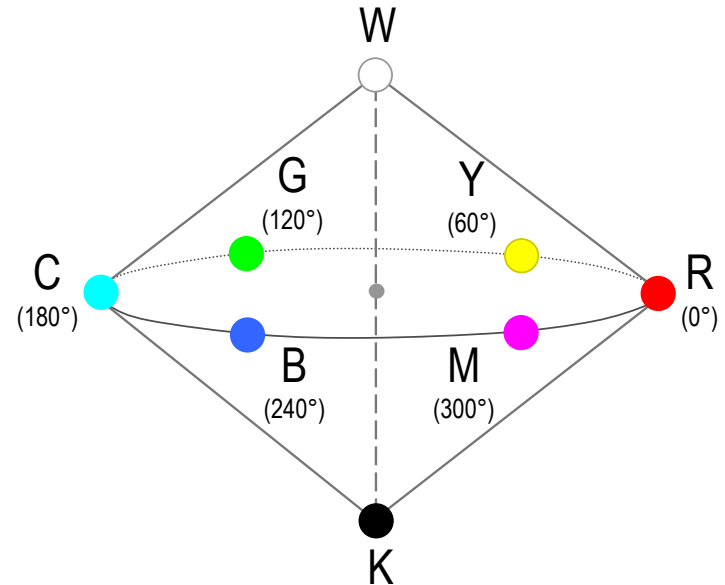
1. ak $S == 0$, $R = V$, $G = V$, $B = V$, koniec
2. inak $H = H / 60$
3. $i = \text{celá časť } H$, $f = \text{zvyšok po delení } H$
4. $p = V * (1 - S)$, $q = V * (1 - (S * f))$, $t = V * (1 - (S * (1 - f)))$
5. ak $i == 0$ $R = V$, $G = t$, $B = p$
6. ak $i == 1$ $R = q$, $G = V$, $B = p$
7. ak $i == 2$ $R = p$, $G = V$, $B = t$
8. ak $i == 3$ $R = p$, $G = q$, $B = V$
9. ak $i == 4$ $R = t$, $G = p$, $B = V$
10. ak $i == 5$ $R = V$, $G = p$, $B = q$



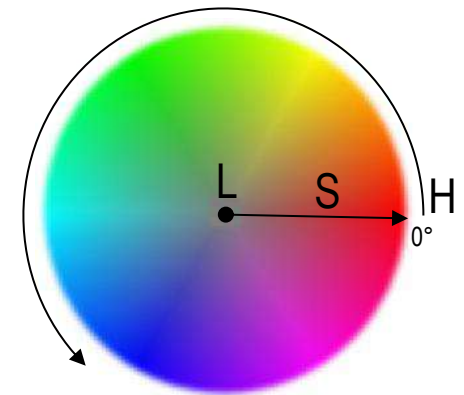
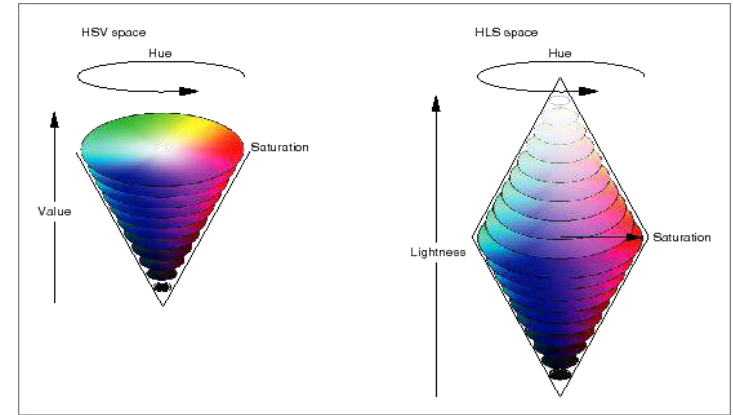
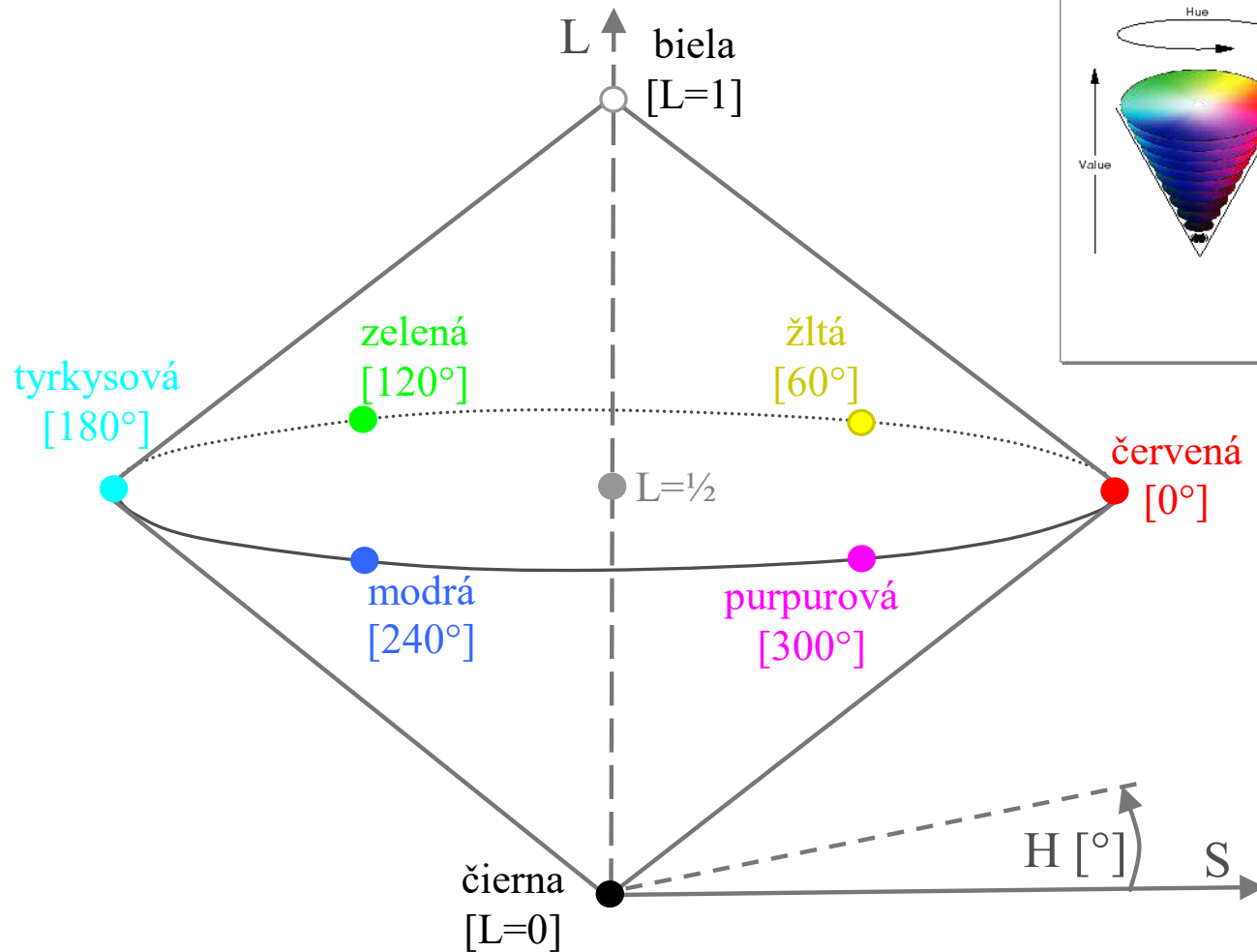
FAREBNÝ MODEL HLS

- Zložky
 - **Hue** – farebný tón
 - **Lightness** – svetlosť
 - **Saturation** – saturácia

- Miešanie: aditívne
- Zmena zložky: uhlová a lineárna



FAREBNÝ MODEL HLS



PREVOD RGB $\rightarrow$ HLS

1. $\text{minim} = \min(R, G, B)$
2. $\text{maxim} = \max(R, G, B)$
3. $L = (\text{minim} + \text{maxim}) / 2$
4. ak $L == 1$, $H = 0$, $S = 0$, koniec //biela farba
5. ak $\text{minim} == \text{maxim}$, $S = 0$, $H = 0$, koniec //odtiene šedej
6. inak $S = (1 - \text{minim}) / (1 - L) - 1$
7. ak $L > 0.5$

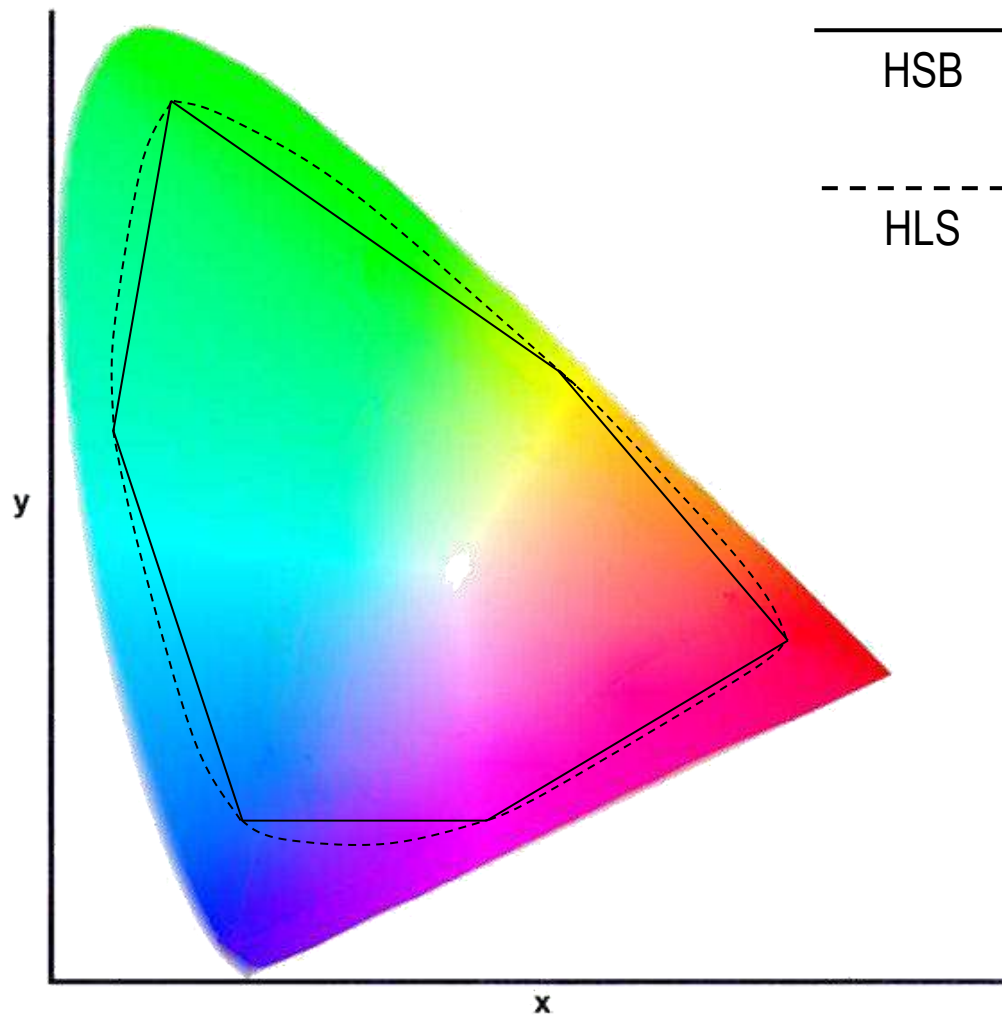
$$R2 = 1 + (R - 1) / (2 - 2L)$$

$$G2 = 1 + (G - 1) / (2 - 2L)$$

$$B2 = 1 + (B - 1) / (2 - 2L)$$
 inak $R2 = R$, $G2 = G$, $B2 = B$
8. ak prevláda zložka R: ak $G2 > B2$, $H = G2 / B2$ inak $H = 6 - B2 / G2$
9. ak prevláda zložka G: ak $R2 > B2$, $H = 2 - R2 / G2$ inak $H = 2 + B2 / G2$
10. ak prevláda zložka B: ak $G2 > R2$, $H = 4 - G2 / B2$ inak $H = 4 + R2 / B2$
11. $H = H * 60$, ak $H < 0$: $H = H + 360$



POROVNANIE HSB A HLS FAREBNÉHO PRIESTORU



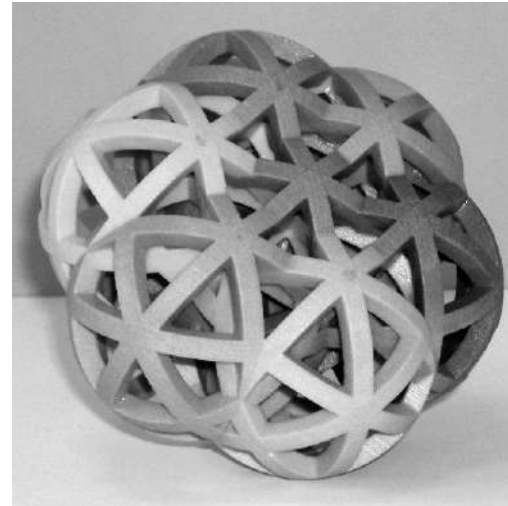
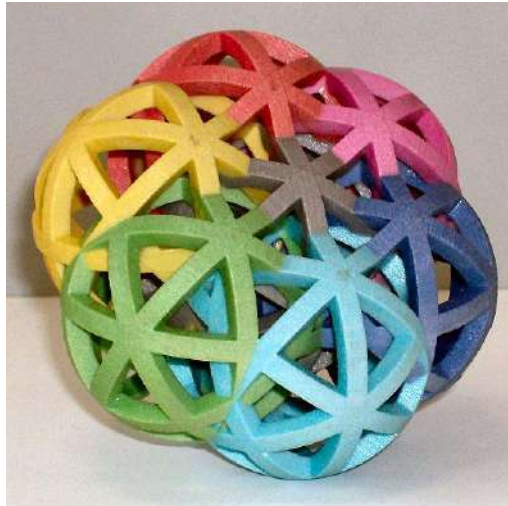
PREVOD RGB DO ODTIEŇOV ŠEDEJ

$$I = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$$

I - je výsledná intenzita (úroveň šedej)

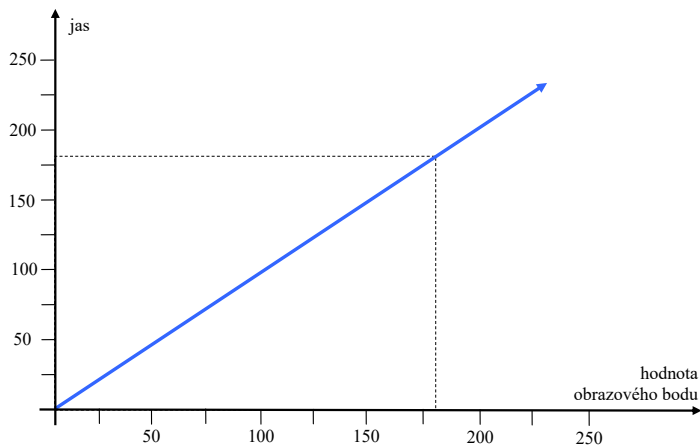
R, G, B - sú základné farebné zložky pôvodnej farby

(zdroj LIRKIS KPI FEI TU Košice)

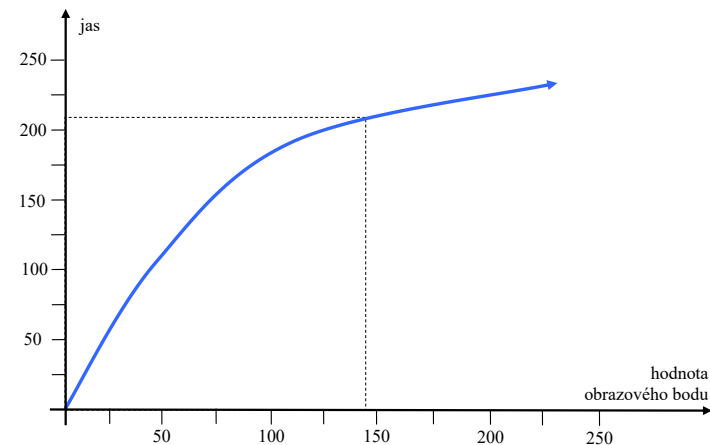


GAMA KOREKCIA

Koeficient *gama*, definuje vzťah medzi číselnou hodnotou obrazového bodu a jeho skutočnou svietivosťou. *Gama korekcia* sa aplikuje preto, lebo ľudské oko má vyššiu citlivosť v tieni ($L=0,5$), kým pri jasnom svetle ($L=1$) alebo v tme ($L=0$) je menej citlivé. Naopak technické prostriedky (senzory, snímače, zobrazovače) vnímajú/emituju svetlo spravidla lineárne. *Gama korekcia* tak pomáha vylepšiť techniku expozície/iluminácie, ktorá vzniká rozdielom medzi technickým spracovaním farieb a prirodzeným ľudským vnímaním.



Ideálna (technická, lineárna) závislosť hodnoty obrazového bodu a jeho jas



Reálna (ľudská, nelineárna) závislosť hodnoty obrazového bodu a jeho jas

MERANIE FARIEB

SPEKTROMETRE A KOLORIMETRE

- **Spektrometer** meria odrazené alebo prechádzajúce svetlo v mnohých bodoch vizuálneho spektra, výsledkom čoho je krivka. Keďže krivka každej farby je jedinečná ako podpis alebo odtlačok prsta, krivka jednoznačne špecifikuje a tým sa identifikuje zhoda farieb. Je to teda druh vedeckého prístroja, ktorý umožňuje skúmať prvkové (chemické) zloženie látky či objektu na báze merania odrazeného svetla respektíve odrazenej vlnovej dĺžky svetla a jeho absorpcie alebo na základe merania vzniknutého svetla, pričom ku vzniku dochádza umelou excitáciou (plazma, iskra, rtg, ...). Proces sa volá spektrometria. Spektrometre sa často využívajú pre získavanie dát o zložení telies najmä v kozmickom výskume. pri výrobe materiálov, riadenie výroby, kontrolách chemického zloženia, testovaní farieb atramentov, kontrole farby náterov či kontrole farby plastov a textílií .
- **Kolorimeter** nie je spektrometer. Kolorimetre sú tristimulačné (trojfiltrové) zariadenia, ktoré využívajú červené, zelené a modré filtre, ktoré napodobňujú reakciu ľudského oka na svetlo a farbu. Kolorimeter meria absorbanciu určitých vlnových dĺžok svetla špecifickým senzorom alebo aj roztokom. V niektorých aplikáciách kontroly kvality predstavujú tieto nástroje najlepší pomer cena/výkon. Kolorimetre nedokážu kompenzovať metamériu (posun vo vzhľade vzorky v dôsledku svetla (svetelného zdroja) použitého na osvetlenie povrchu). Nedokážu to preto, lebo kolorimetre používajú spravidla jeden typ referenčného svetla (ako je žiarovka alebo pulzný xenón) a nezaznamenávajú spektrálnu odrazivosť média. Spektrometre dokážu kompenzovať tento posun, vďaka čomu sú spektrometre vynikajúcou voľbou pre presné a opakovateľné meranie farieb. Kolorimeter sa používa aj na meranie/kalibráciu monitorov/displejov.



ERGONOMIA FARIEB

SYMBOLIKA A „PSYCHOLÓGIA“ FARIEB

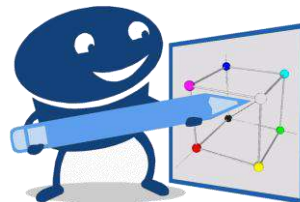
- **biela** symbolizuje čistotu a nepoškvrnenosť; je to veselá farba.
- **čierna** symbolizuje smútok; je to pasívna farba.
- **modrá** symbolizuje vernosť a diaľku; pôsobí zdržanlivo, uvoľňujúco, kludne a vážne; je to studená a pasívna farba.
- **červená** symbolizuje lásku; pôsobí energicky, vzrušujúco až bombasticky; je to teplá, aktívna a dominantná farba.
- **zelená** symbolizuje nádej; pôsobí osviežujúco, ukludňujúco; je to pasívna, neutrálna farba.
- **šedá** symbolizuje nerozhodnosť; je to pasívna, neenergetická a nedominantná farba.
- **žltá** symbolizuje inteligenciu; pôsobí veselo, vzrušujúco a mocne; je to aktívna, teplá a dominantná farba.



ERGONÓMIA FARIEB

POUŽITIE FARIEB V PG

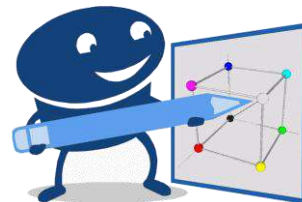
- počet rôznych farieb
- podobnosť farieb
- zoskupenie spoločnou farbou pozadia
- jas a intenzita farieb
- stupeň farebnej zmeny
- "teplota" farieb



ERGONOMIA FARIEB

NIEKTORÉ PRAVIDLÁ POUŽITIA FARIEB V PG

- výhodné je pre dosiahnutie veselosti, rozhodnosti a vyrovnanosti použiť pestrú paletu farieb. Ale ako vždy platí, všetko s mierou.
- použitie silných farebných kontrastov typu svetlý-tmavý budí pozornosť. Vďaka rôznej svetlosti a sýtosti sa charakter farby môže meniť.
- na získanie dojmu pôsobenia priestoru sa použije kontrast studených a teplých farieb (červená/zelená, modrá/žltá, fialová/oranžová). Je to dané tým, že teplé farby vystupujú do popredia a studené ustupujú do úzadia.
- na vytvorenie akcentov (dominantných častí) je možné použiť kombináciu žiarivých (červená, oranžová) a kalných (zelená, modrá) farieb.
- nakoniec aj používanie monochromatickej palety (len svetlé alebo tmavé odtiene tej istej farby) alebo používanie príbuzných farieb vytvára niekedy decentný vzhľad.



VPLYV FARBY NA VNÍMANIE VIRTUÁLNEHO SVETA

SYSTÉM ZMIEŠANEJ REALITY LIRKIS

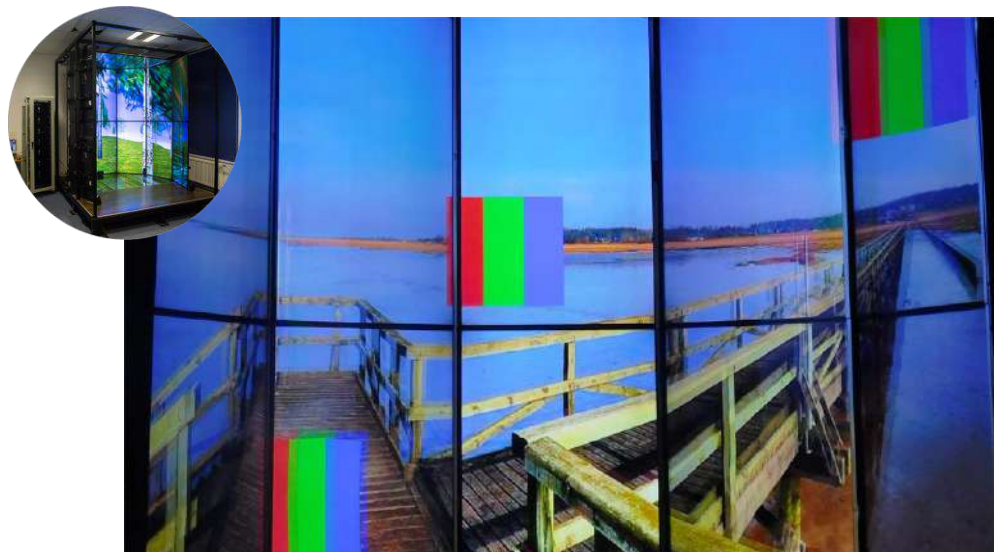


(zdroj LIRKIS KPI FEI TU Košice)



TRÉNING A POSILNENIE FAREBNÉHO VIDENIA V SYSTÉME VIRTUÁLNEJ JASKYNE LIRKIS CAVE

Virtuálna scéna je viac škálovaná v odtieňoch základných farieb (R, G, B) a umožňuje pri tréningovom procese vymeniť jednotlivé farby. Najbežnejšou metódou pri virtuálnom tréningu je zmena farby predmetov na výrazné farby, ktoré pôsobia na pozornosť a podporujú aktiváciu príslušných častí ľudského mozgu. Použiteľné je aj prostredie so skutočným vnímaním a farebnými značkami v rôznych častiach scény



(zdroj LIRKIS KPI FEI TU Košice)

Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

GRAFICKÁ INFORMÁCIA A JEJ SPRACOVANIE

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 02

© 2025

VRSTVY VIZUALIZAČNÉHO PROCESU

1. Definovanie/formalizácia/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



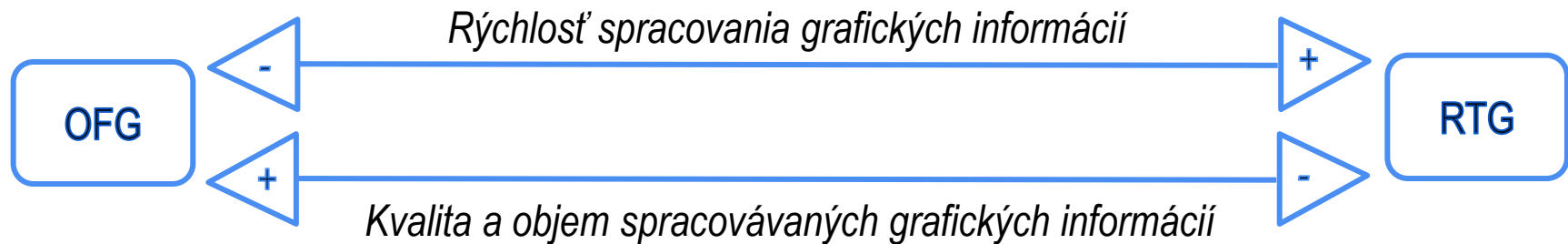
GRAFICKÁ INFORMÁCIA (OBJEKTY, ENTITY)

1. Typy
2. Primitíva
 1. implementácia primitív
 2. filtračné a rozptyľovacie metódy
3. Popis objektu
4. Reprezentácia objektu
5. Priestor objektu



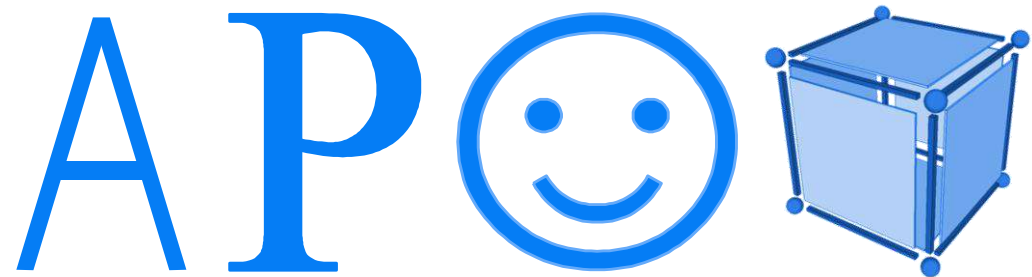
GRAFICKÁ INFORMÁCIA (SPRACOVANIE)

1. Grafická informácia spracovávaná mimo reálneho času (*OFG* (off-line graphics))
2. Grafická informácia spracovávaná v reálnom čase (*RTG* (real-time graphics))

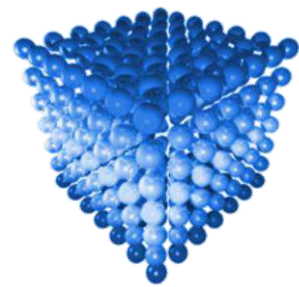
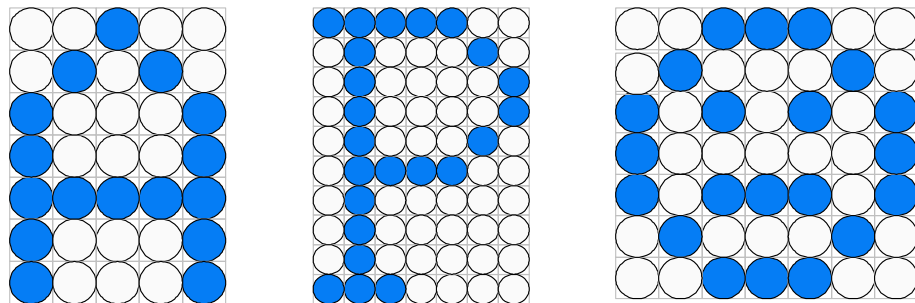


GRAFICKÁ INFORMÁCIA - TYP

- Vektorová
(spojitý priestor, rôzne entity)



- Rastrová
(nespojité priestor, len body)



GRAFICKÁ INFORMÁCIA

POUŽITIE VEKTOROVEJ GRAFIKY

- Schémy
- Grafy
- Konštruovanie
- Dizajn
- Diagramy
- Reklama
- Mapy
- Piktogramy
- ...



GRAFICKÁ INFORMÁCIA

POUŽITIE RASTROVEJ GRAFIKY

- Obrázky
- Fotografie
- Filmy
- Komunikácia
- Skenovanie
- Televízia
- Biometria
- ...

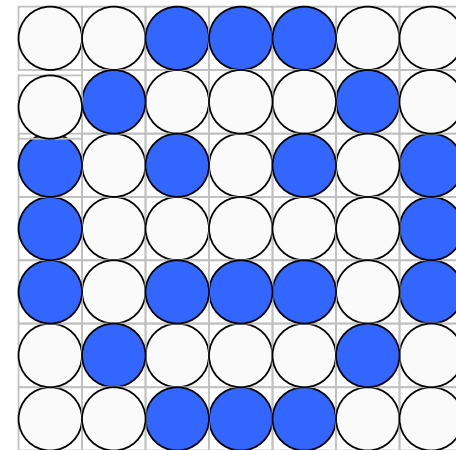


GRAFICKÁ INFORMÁCIA

TRANSFORMÁCIE TYPOV



Vektorový typ



Rastrový typ

GRAFICKÉ OBJEKTY

1. Vektorové

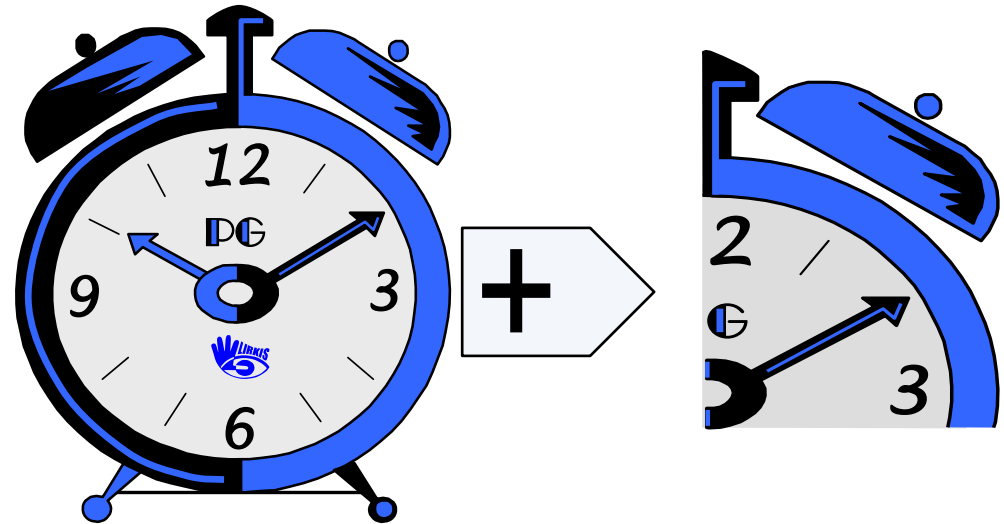


2. Rastrové

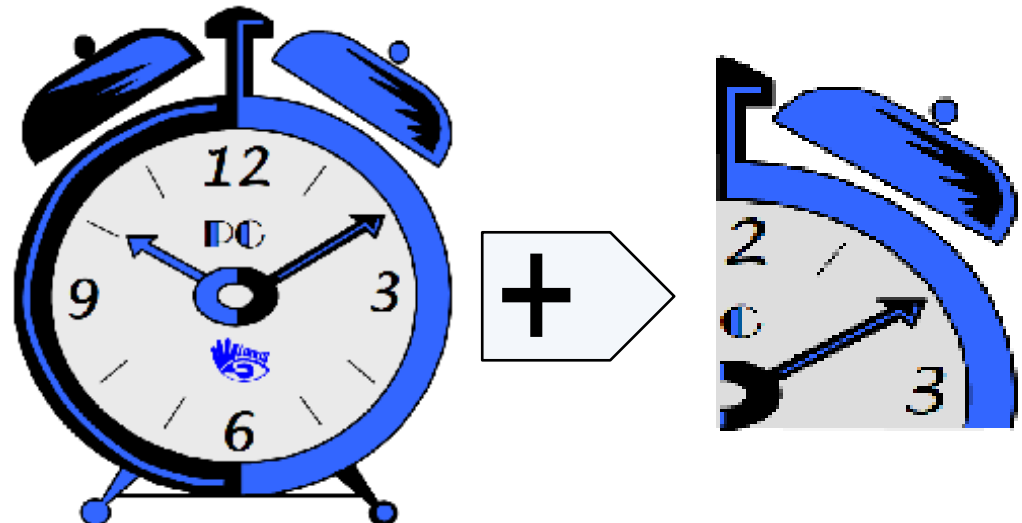


ZVÄČŠENIE OBJEKTOV

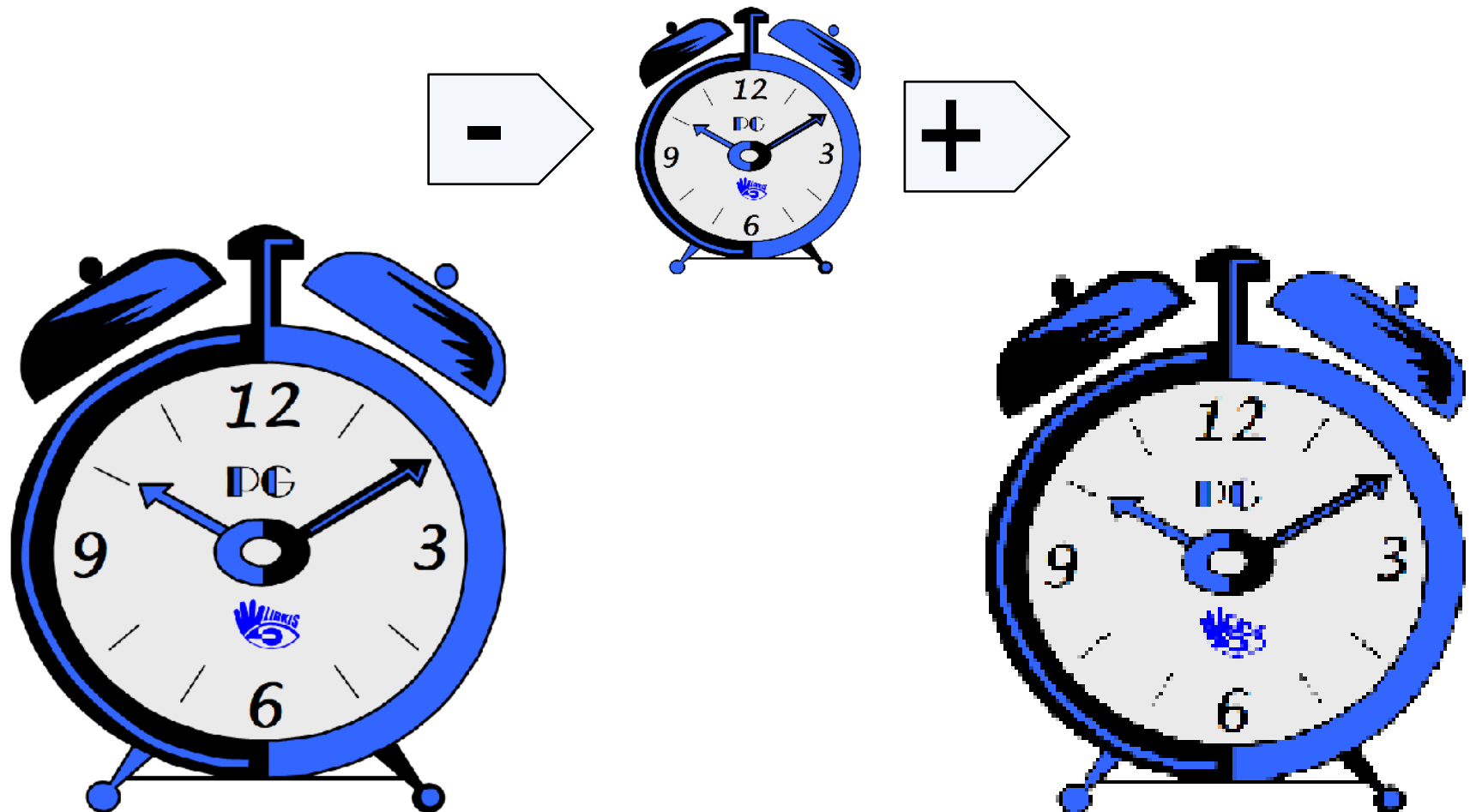
1. Vektorové



2. Rastrové



STRATA INFORMÁCIE PRI RASTRI

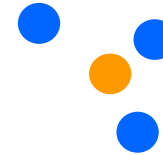


GRAFICKÉ PRIMITÍVA

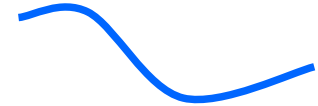
1. Bod
2. Sled bodov
3. Krivka
4. Lomená čiara
5. Grafický text
6. Plocha
7. Vyplnená oblasť
8. Výplňový vzor
9. Všeobecný grafický prvok



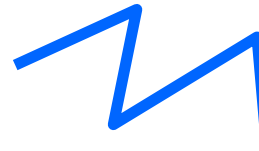
1



2



3



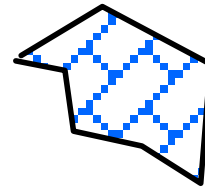
4



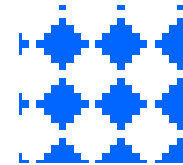
5



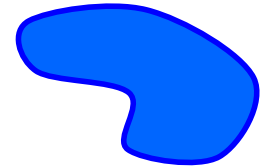
6



7



8



9

GRAFICKÉ PRIMITÍVA - ATRIBÚTY

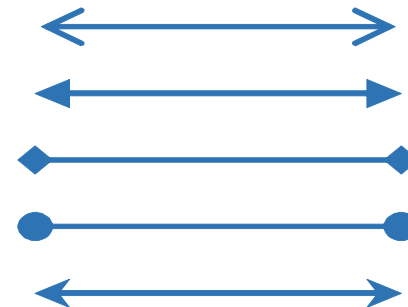
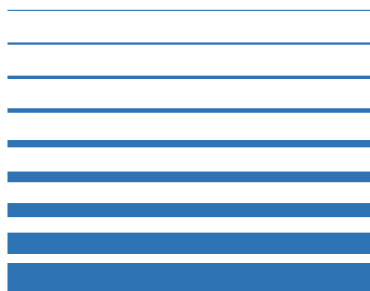
Konečný tvar primitív je možné riadiť ich atribútmi. Medzi základné atribúty z pohľadu počítačovej grafiky zaradujeme najmä:

1. Farba
2. Typ (napr. čiar, písma a pod.)
3. Hrúbka (napr. čiar, písma a pod.)
4. Poloha (napr. písma)
5. Smer vykreslenia (napr. horizontálny, vertikálny atď.)

GRAFICKÉ PRIMITÍVA - ATRIBÚTY

Atribúty môžu byť jednotlivým elementom priradené:

1. *konvenčne* alebo tiež *individuálne*, *pevne*, čo vedie niekedy k nekompatibilite na rôznych zobrazovačoch.
2. *symbolicky*, najčastejšie formou kódu. Vtedy hovoríme o *viazaných* (bundled) atribútoch. Tieto sú vzhľadom na zobrazovacie zariadenie nezávislé.



Príklady rôznych atribútov čiar

BOD

Bod je z pohľadu počítačovej grafiky chápaný ako elementárny (atomárny) objekt.

Základnými atribútmi bodu z pohľadu počítačovej grafiky, sú: jeho *poloha* a *farba* príp. *čas* (v prípade heterogénnej štruktúry dimenzie priestoru, ktorej je bod prvkom)

Niektoré typy bodov:

- **Pixel** – obrazový bod, ktorý je charakterizovaný dvomi súradnicami polohy a svojou farbou. Z pohľadu PG je pixel chápaný ako najmenšia jednotka digitálnej rastrovej grafiky.
- **Voxel** – objemový bod, charakterizovaný tromi polohovými súradnicami a svojou farbou.
- **Texel** – bod textúry, okrem polohy v rámci určitej súradnicovej sústavy má ešte definovanú aj polohu v rámci výplňového vzoru (textúry) a reláciu svojho priradenia vyplňanej oblasti.

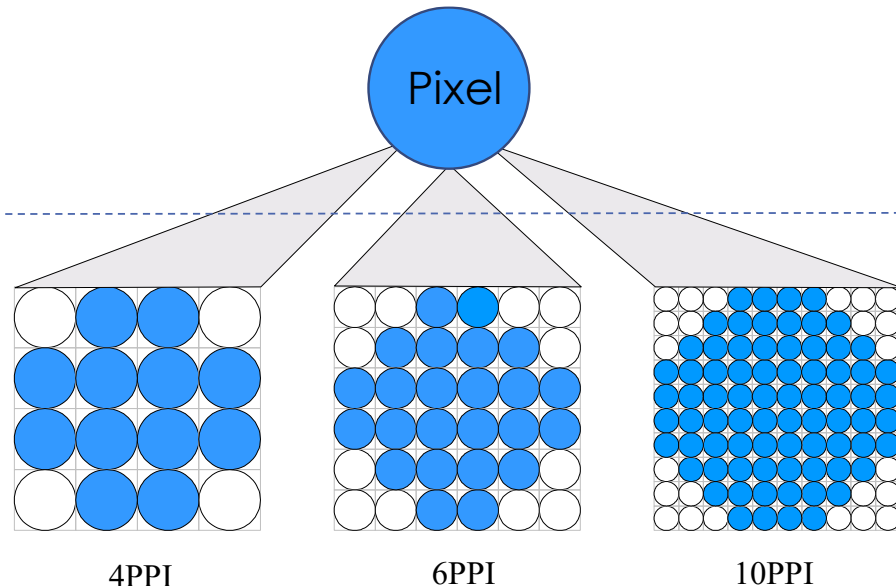
FYZICKÉ VYTVORENIE BODU, ROZLIŠENIE

Na rastrovom zobrazovacom zariadení (napr. na monitore) je pixel vytvorený zobrazením (rozsvietením) istej množiny fyzických bodov výstupnej jednotky zariadenia.

Jemnosť (narastajúca hustota a veľkosť) fyzických bodov ovplyvňuje kvalitu zobrazenia. Udáva sa podľa zariadenia v *PPI* (*Pixel Per Inch*) alebo *DPI* (*Dot Per Inch*)

Pixel na logickej
atomárnej úrovni

Fyzicky reprezentovaný
pixel na výstupnom
rastrovom zariadení



SLED BODOV - POLYMARKER

Je rozširujúcim prvkom a priamo nadväzuje na bod.

Polymarker definuje logicky zviazanú *množinu bodov* na základe určitej *relácie/cii* (neprihliada sa na lineárnosť alebo nelineárnosť relácie/funkcie) medzi prislúchajúcimi atribútmi týchto bodov. Pri akejkoľvek operácii (napr. presúvaní) sa táto operácia deje nad všetkými bodmi spolu.

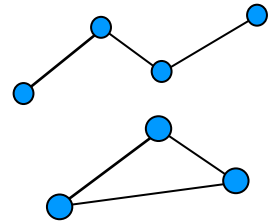
Relácia medzi atribútmi bodov polymarkra môže byť:

- **homogénna** - je relácia definovaná medzi rovnakými atribútmi jednotlivých bodov polymarkra. Homogénna relácia, napr. medzi atribútmi polohy.
- **heterogénna** - je definovaná medzi rôznymi atribútmi jednotlivých bodov polymarkra, napr. farba niektorého bodu je závislá od polohy druhého. Platí že, ak je v polymarkri aspoň jedna relácia heterogénneho charakteru, potom celý polymarker je heterogénny.

KRIVKA, LOMENÁ ČIARA

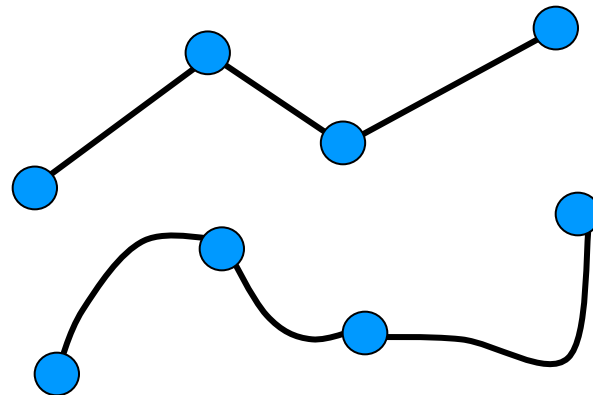
1D útvar definovaný vrcholmi a hranami:

- neuzavretý (acyklický) – krivka, polyline
- uzavretý (cyklický) – plocha, polygón



Podľa typov hrán (segmentov) krivky/čiary:

- lineárny
- nelineárny

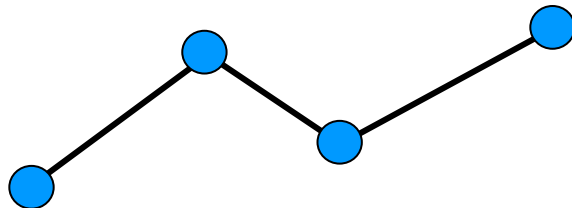


KRIVKA, LOMENÁ ČIARA

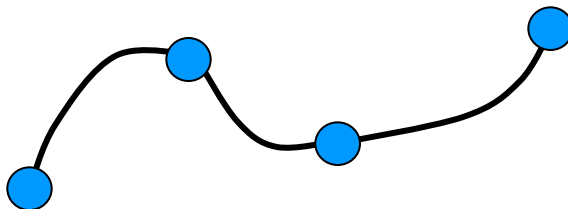
Podľa vplyvu vrcholov na tvar krivky/čiary:

- **Interpoláčné** (vrcholy sú súčasťou krivky), typovo: lineárna a nelineárna.
- **Aproximačné** (vrcholy nemusia byť súčasťou krivky, ale vplývajú na jej tvar), typovo: lineárna a nelineárna.

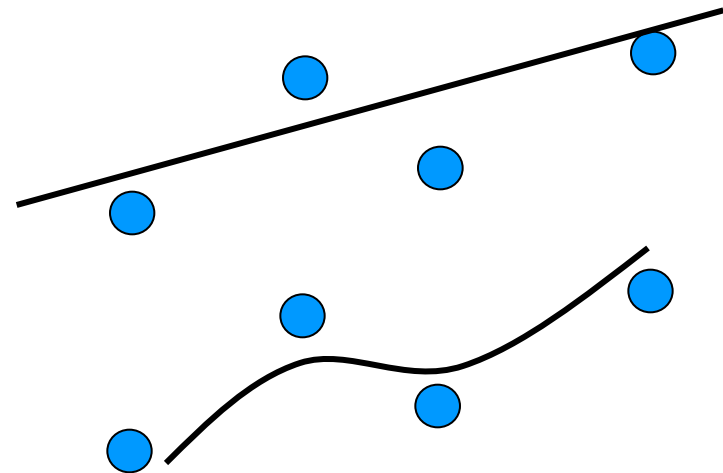
lineárna



nelineárna



interpolácia



aproximácia

ZÁKLADNÉ SPRACOVÁVANÉ GRAFICKÉ PRIMITÍVA

- Úsečka (priamka)
- Kružnica
- Elipsa
- Plocha

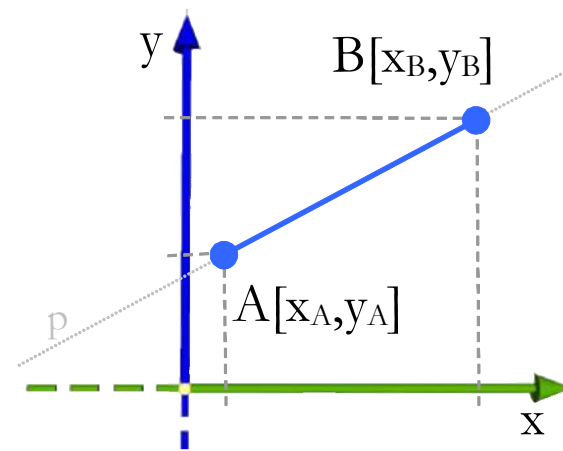


ÚSEČKA

$$y = k \cdot x + c$$

kde:

- y - súradnica bodu na osi y
- x - súradnica bodu na osi x
- k - smernica priamky, na ktorej leží úsečka
- c - posun na osi y



Ak máme dva body $A[x_A, y_A]$ a $B[x_B, y_B]$, ktoré sú koncovými bodmi úsečky, potom základné koeficienty vypočítame nasledovne:

$$k = \frac{y_B - y_A}{x_B - x_A} \quad \text{a} \quad c = \frac{(x_B y_A - x_A y_B)}{(x_B - x_A)}$$

ALGORITMY KRESLENIA ÚSEČKY (RASTROVEJ NÁHRADY)

- Algoritmus založený na výpočte oboch súradníc (analytický parametrický var)
- DDA - digital differential analyzer
- Bresenhamov algoritmus



DDA ALGORITMUS

prírastkový algoritmus

(DDA - digital differential analyzer)

založený na postupnom pripočítavaní konštantných prírastkov k obom súradniciam x a y .

Rozlišujeme výpočet pre priamku:

- so smernicou menšou ako 1
- so smernicou väčšou ako 1

DDA ALGORITMUS

$$k = \frac{y_B - y_A}{x_B - x_A} \quad \text{a} \quad c = \frac{(x_B y_A - x_A y_B)}{(x_B - x_A)}$$

$$\begin{aligned} dx &= |x_B - x_A| \\ dy &= |y_B - y_A| \end{aligned} \Rightarrow \text{pocet\_krokov} = \max(dx, dy) \Rightarrow \begin{aligned} px &= \frac{dx}{\text{pocet\_krokov}} \\ py &= \frac{dy}{\text{pocet\_krokov}} \end{aligned}$$



$$y_{i+1} = y_i + py$$

$$x_{i+1} = x_i + px$$



DDA ALGORITMUS (IMPLEMENTÁCIA)

```

dda_line(xA, yA, xB, yB){

    body_priamky = [];

    dx = xB - xA;
    dy = yB - yA;

    pocet_krokov = int(max(abs(dx), abs(dy)));

    px = dx / pocet_krokov ;
    py = dy / pocet_krokov ;

    x = xA;
    y = yA;

    for i( 0, pocet_krokov, i++){
        body_priamky.append((round(x), round(y)));
        x += px;
        y += py;
    }

    return body_priamky;
}

```



BRESENHAMOV ALGORITMUS

veľmi efektívny algoritmus generovania bodov na úsečke.
Nachádza body ležiace najbližšie danej skutočnej úsečke
na základe hodnoty predikčného chybového člena Err_D .

Rozlišujeme výpočet pre priamku:

- so smernicou menšou ako 1
- so smernicou väčšou ako 1
- v rôznych kvadrantoch



Dr. Jack Elton Bresenham

BRESENHAMOV ALGORITMUS

$$dx = |x_B - x_A|$$

$$dy = |y_B - y_A|$$

ak $(x_A < x_B)$ $krok\_sx = 1$ inak $krok\_sx = -1$

ak $(y_A < y_B)$ $krok\_sy = 1$ inak $krok\_sy = -1$

Úprava krokov podľa
smernice a kvadrantu

$$Err_D = dy - dx$$

$$x_i = x_A \quad y_i = y_A$$

Inicializácia chybového člena

Inicializácia bodov úsečky

pokým $x_i \neq x_B \wedge y_i \neq y_B$

Hlavný cyklus

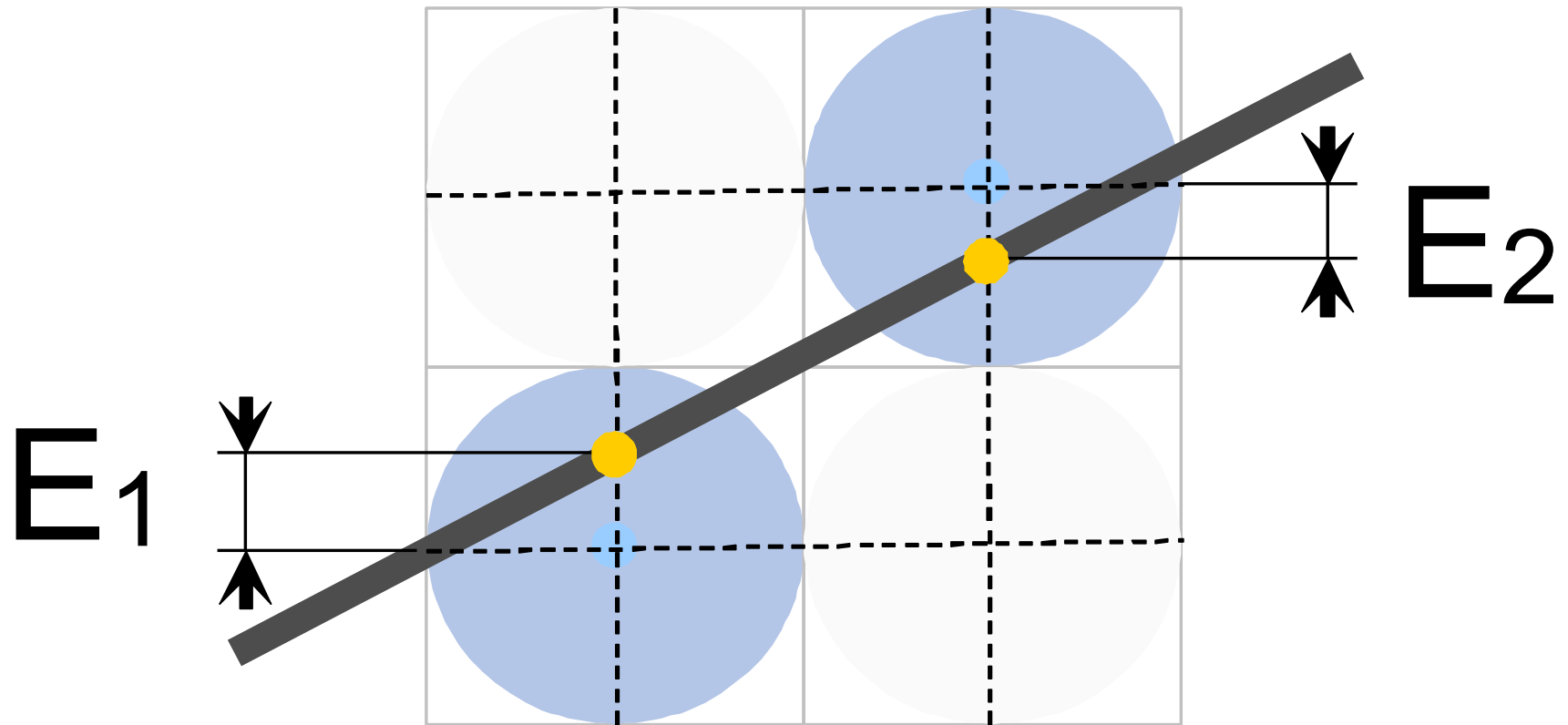
$$i = i + 1$$

$$Err_{D2} = 2 * Err_D$$

ak $(Err_{D2} > dy)$ $Err_D = Err_D - dy$ a $x_i = x_{i-1} + krok\_sx$

ak $(Err_{D2} < dx)$ $Err_D = Err_D + dx$ a $y_i = y_{i-1} + krok\_sy$

BRESENHAMOV ALGORITMUS



Stanovenie chybového člena Err_D pri Bresenhamovom algoritme (v y-smere)

BRESENHAMOV ALGORITMUS (IMPLEMENTÁCIA)

```

bresenham_line(xA, yA, xB, yB){

    body_priamky = [];

    dx = abs(xB - xA);
    dy = abs(yB - yA);

    if (xA < xB) sx = 1 else -1; // krok x podľa smernice a kvadrantu
    if (yA < yB) sy = 1 else -1; // krok y podľa smernice a kvadrantu

    errd = dx - dy;

    while (xA != xB and yA != yB) {

        body_priamky.append((xA, yA));
        errd2 = 2 * err;

        if ( errd2 > -dy ) { errd -= dy; xA += sx; }
        if ( errd2 <  dx ) { errd += dx; yA += sy; }

    }

    return body_priamky;

}

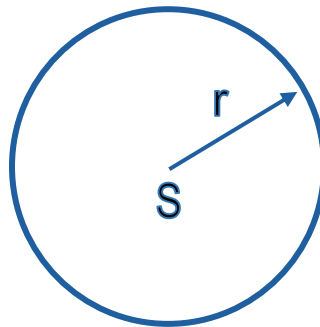
```



KRUŽNICA

množina bodov rovnako vzdialených od stredového bodu S.

$$y = y_s \pm \sqrt{r^2 - (x - x_s)^2}$$



KRUŽNICA

- Algoritmus kreslenia kružnice na základe parametrického analytického vyjadrenia

$$\begin{aligned} x &= x_s + r \cdot \cos(u) \\ y &= y_s + r \cdot \sin(u) \end{aligned} \quad u \in \langle 0, 360^\circ \rangle$$

- Algoritmus kreslenia kružnice podľa predikcie chyby

$$x_1 = 0$$

$$y_1 = r$$

Poloha:

$$x_{i+1} = x_i + 1$$

y_{i+1} sa stanoví predikciou chyby E_{i+1}

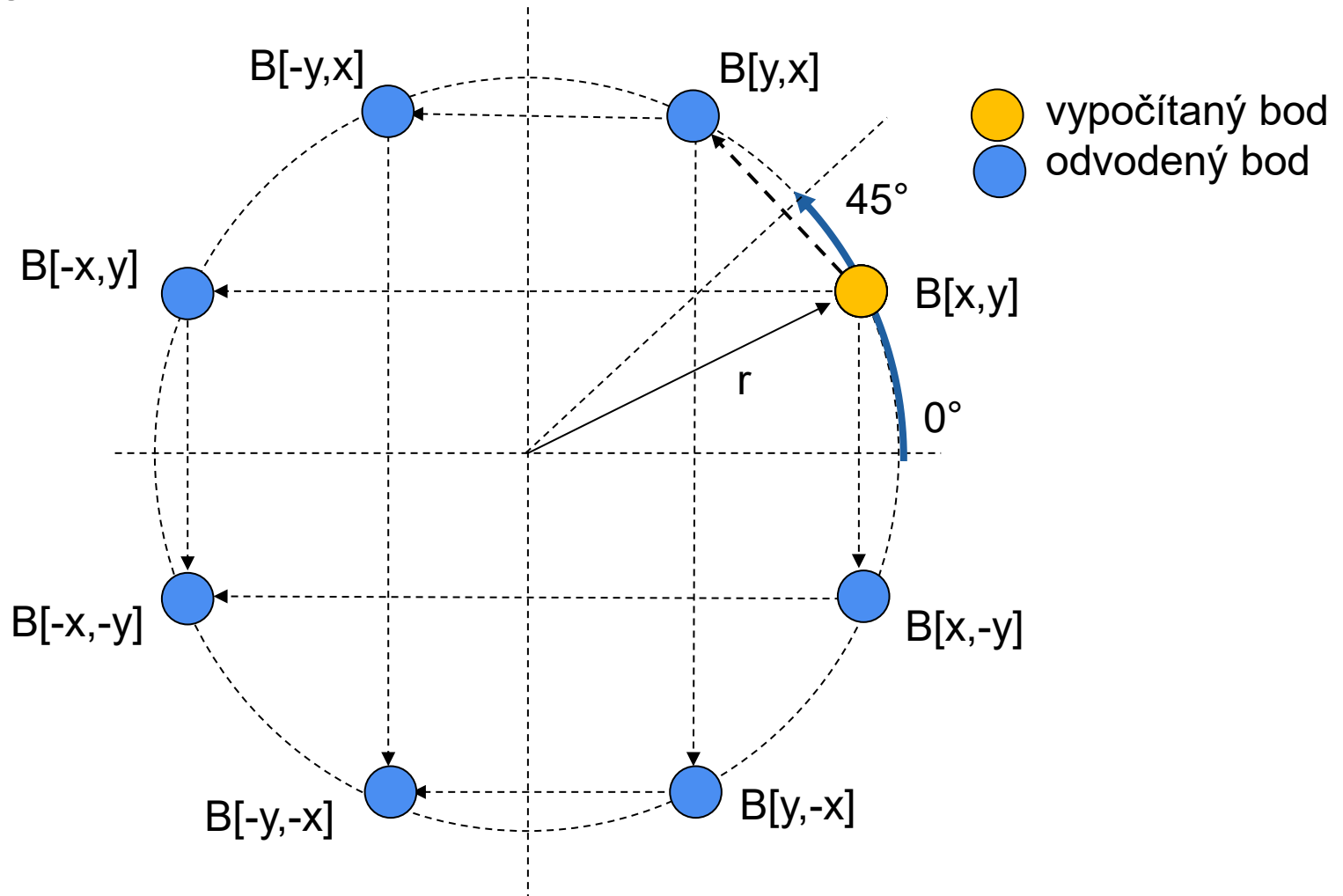
Počiatočná predikcia chyby:

$$E_1 = 1 - r$$

$$E_i < 0 \Rightarrow E_{i+1} = E_i + 2 \cdot x_i + 3$$

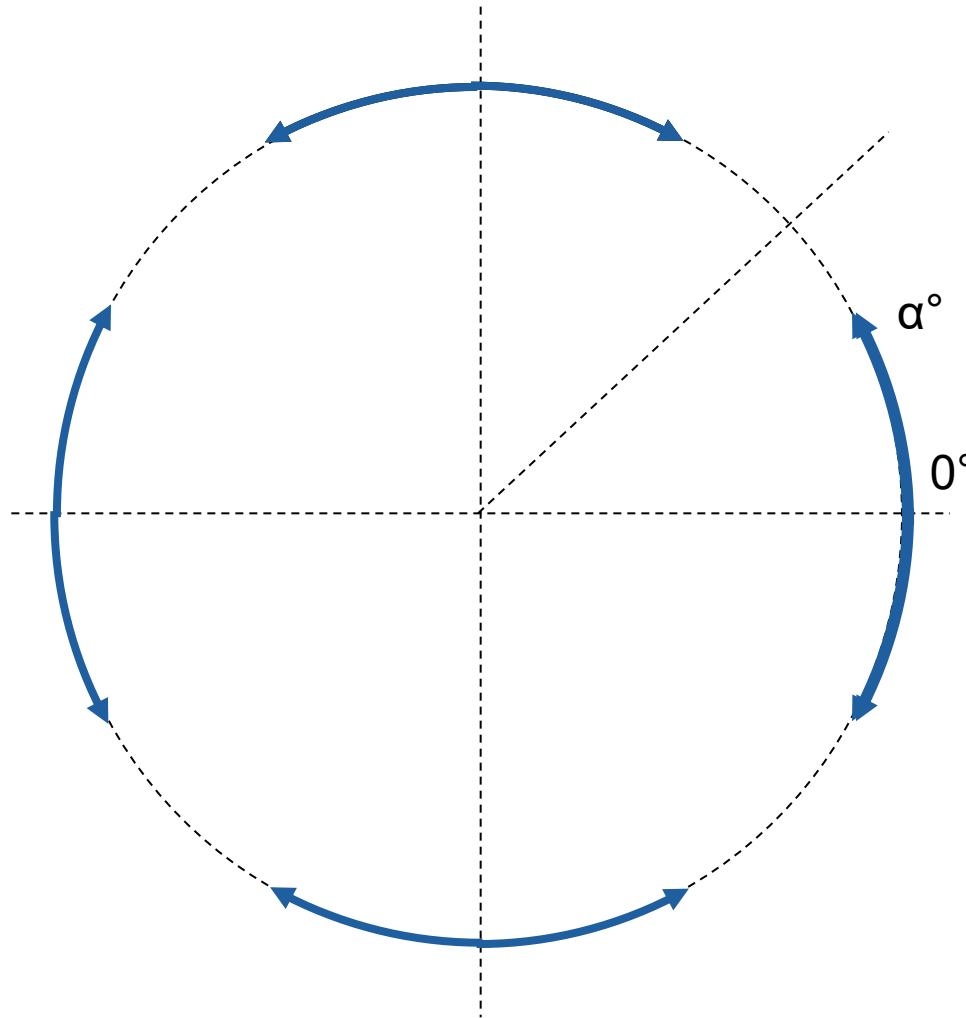
$$E_i \geq 0 \Rightarrow E_{i+1} = E_i + 2 \cdot x_i + 5 - 2 \cdot y_i$$

KRUŽNICA



Kreslenie kružnice využitím osovej súmernosti

KRUŽNICA

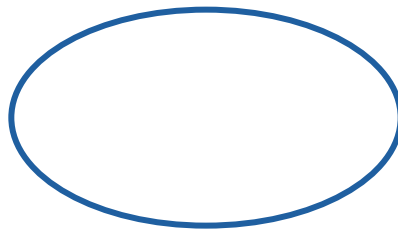


Kreslenie kružnice využitím osovej súmernosti

ELIPSA

množina bodov rovnako vzdialených od dvoch ohnísk

$$\frac{(x - x_s)^2}{a^2} + \frac{(y - y_s)^2}{b^2} = 1$$



ELIPSA

- Algoritmus kreslenia elipsy na základe parametrického analytického vyjadrenia

$$\begin{aligned}
 x &= x_s + a \cdot \cos(u) \\
 y &= y_s + b \cdot \sin(u)
 \end{aligned}
 \quad u \in \langle 0, 360^\circ \rangle$$

- Algoritmus kreslenia elipsy podľa predikcie chyby

$$x_1 = 0$$

$$y_1 = r$$

Poloha:

$$x_{i+1} = x_i + 1$$

y_{i+1} sa stanoví predikciou chyby E_{i+1}

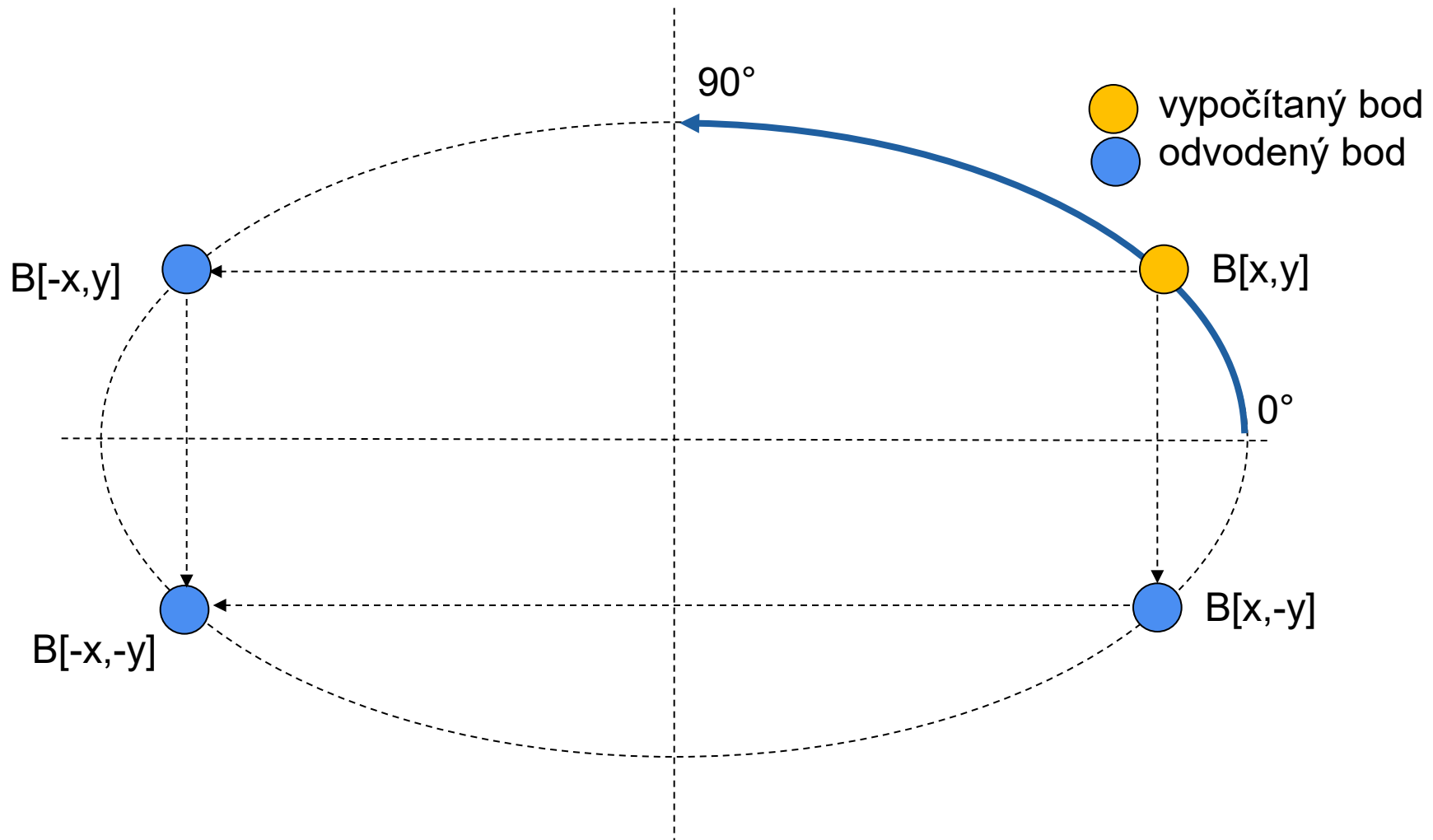
Počiatočná predikcia chyby:

$$E_1 = b^2 - b \cdot a^2 + \frac{a^2}{4}$$

$$E_i < 0 \Rightarrow E_{i+1} = E_i + b^2 \cdot (2 \cdot x_i + 1)$$

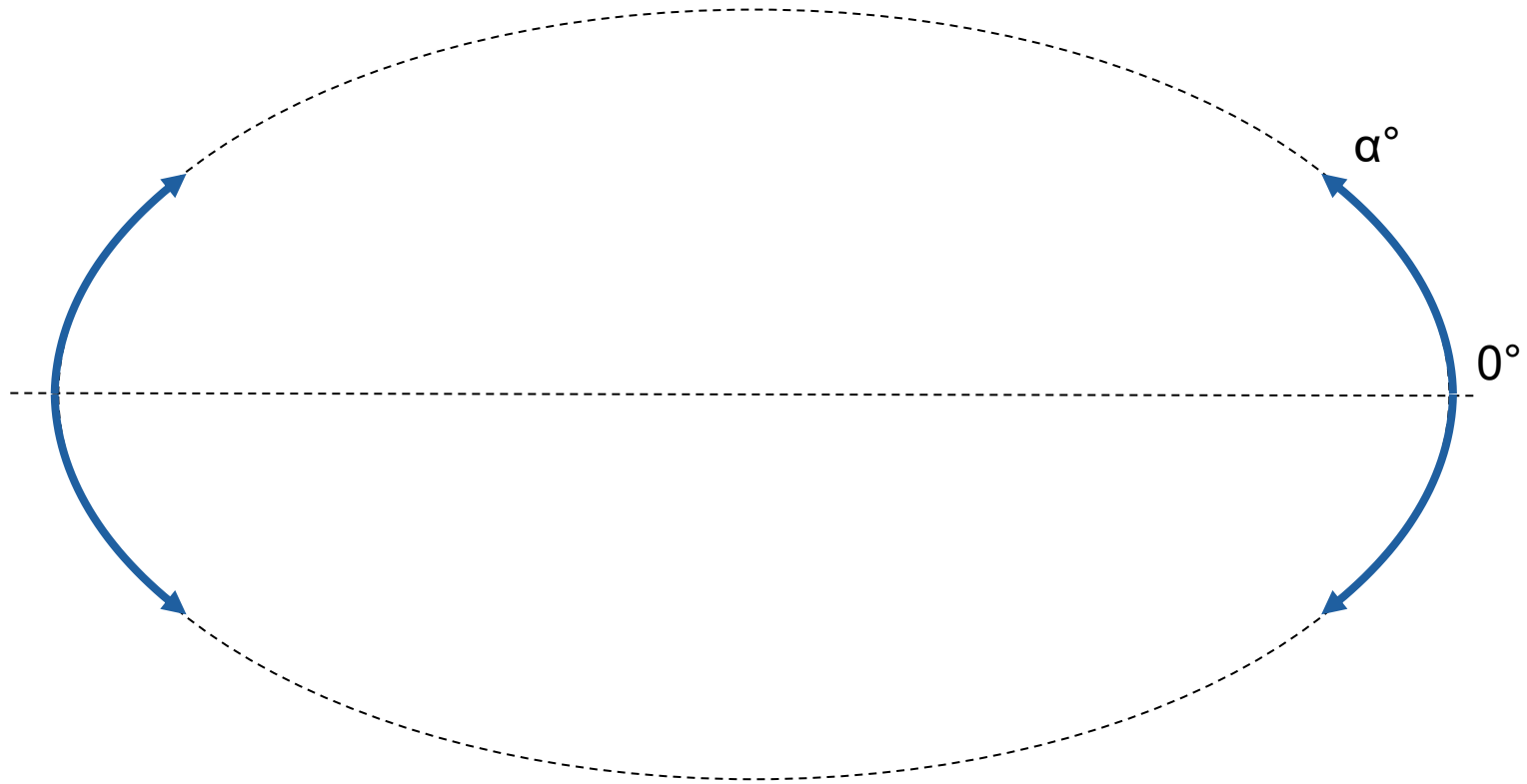
$$E_i \geq 0 \Rightarrow E_{i+1} = E_i + b^2 \cdot (2 \cdot x_i + 1) - 2 \cdot a^2 \cdot y_i$$

ELIPSA



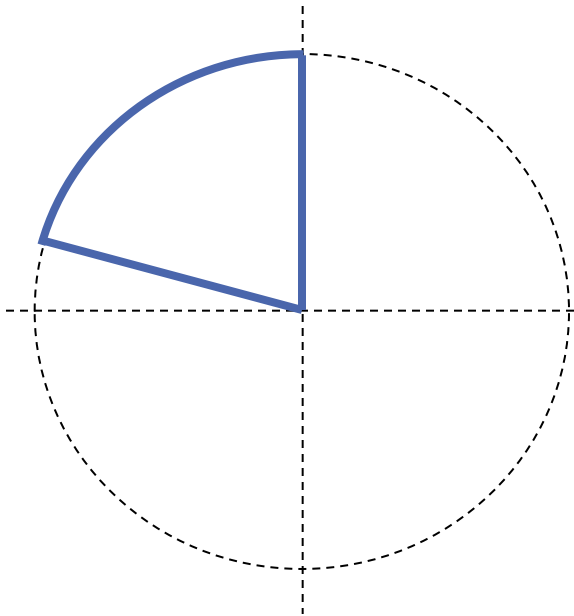
Kreslenie elipsy využitím osovej súmernosti

ELIPSA

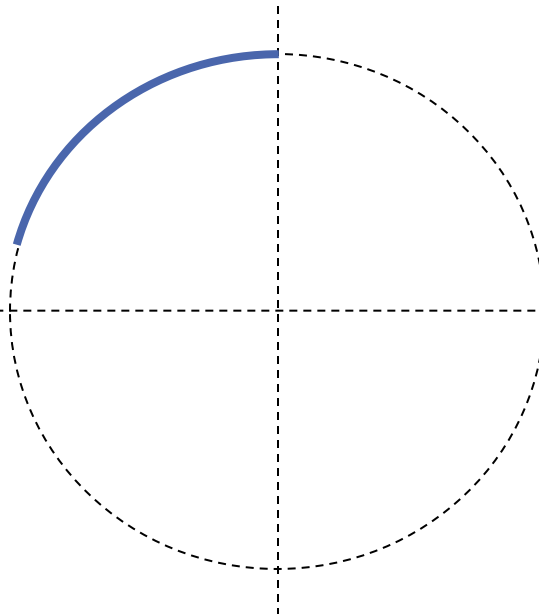


Kreslenie elipsy využitím osovej súmernosti

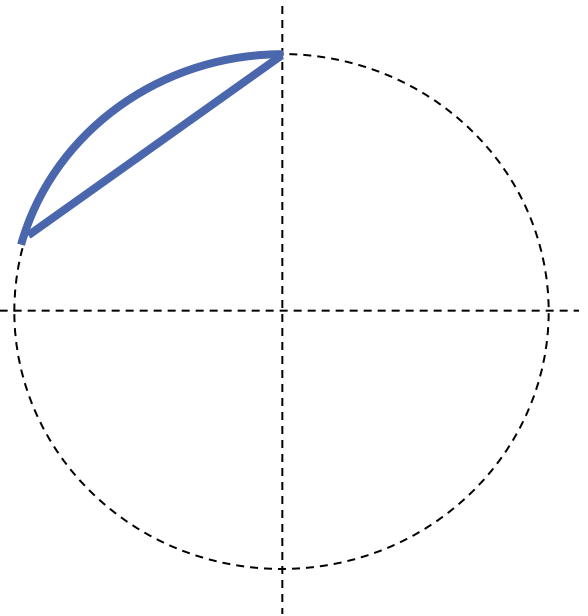
KRUHOVÉ (ELIPTICKÉ) OBLASTI



výsek

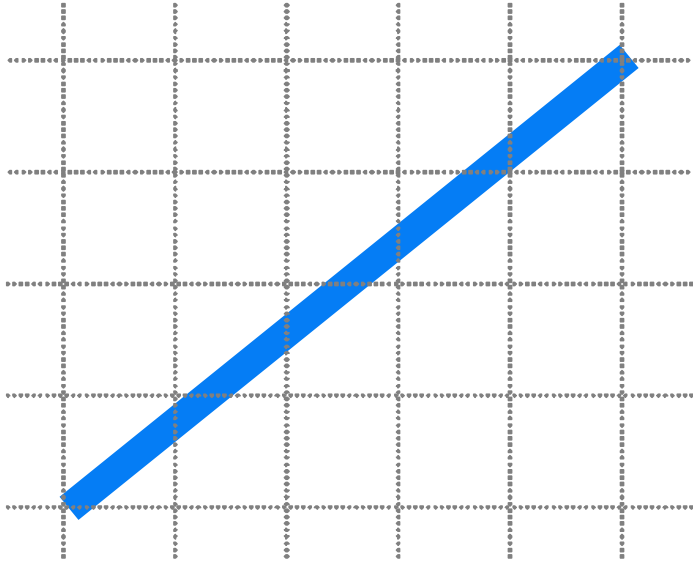


oblúk



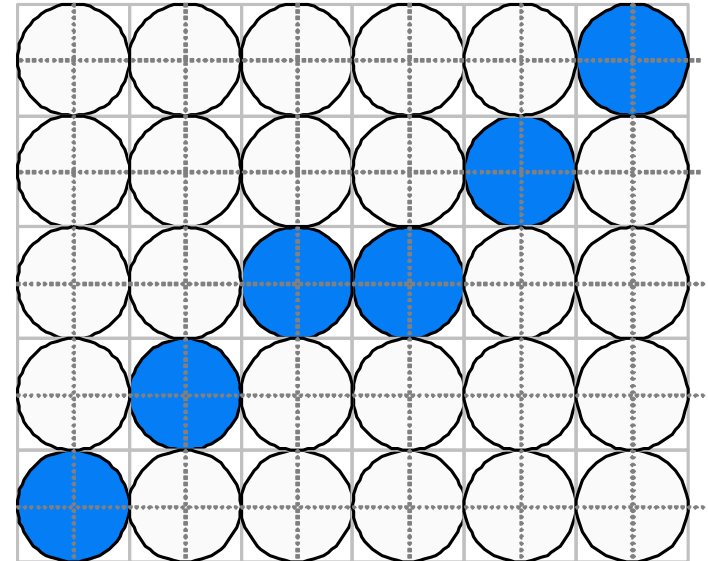
odsek

ANTIALIASING



úsečka

a jej



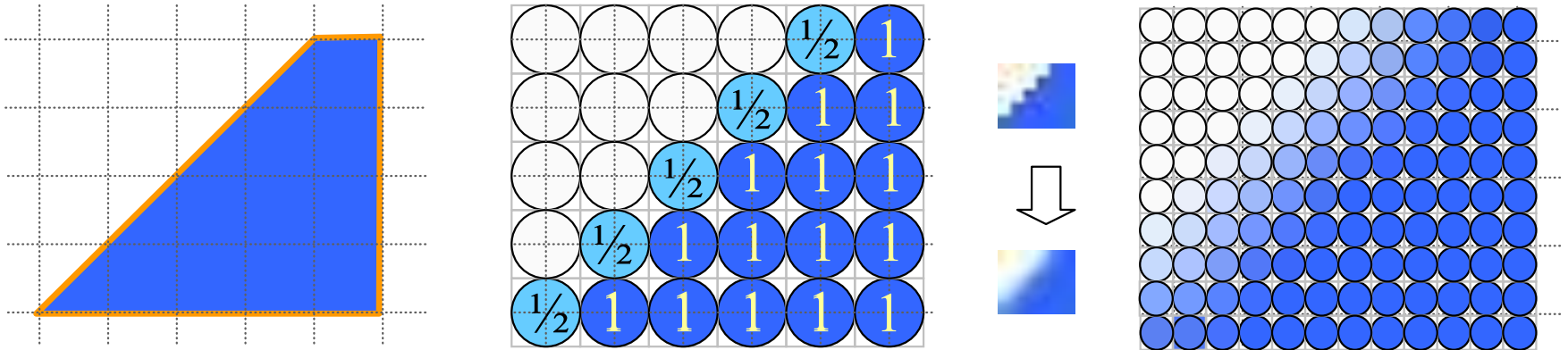
rastrová náhrada
(alias)

ANTIALIASING

Prípady:

- schodovité zobrazovanie priamych rovných čiar a hraníc polygónov na rastrových displejoch.
- zobrazovaný objekt menší ako veľkosť pixelu (obrazového bodu) alebo pri veľmi tenkých čiarach. Malé objekty potom nie sú vôbec zobrazené alebo napr. tenké čiary nie sú opticky hladké resp. ucelené (sú zobrazené ako nepravidelná postupnosť bodov).
- pri zobrazení zložitejšej scény s blízkymi detailmi (napr. pri generovaní základných obrázkov z raytracingu). Tieto detaily sú buď potlačené alebo skreslené tak, že nie je možné rozoznať ich pôvodný tvar. (LOD – Level of Detail)

ANTIALIASING



Častokrát sa používajú miesto *antialiasingu* aj niektoré rozptyľovacie metódy.

FILTROVACIE A ROZPTYĽOVACIE METÓDY

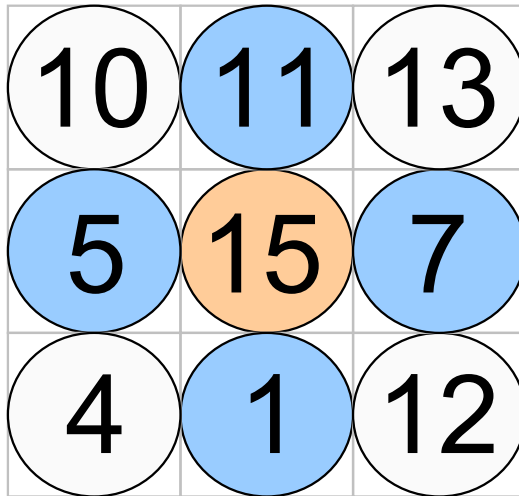
Určenie farby:

- náhodne
- *spriemernením* bodov oblasti alebo použitím *mediánovej* funkcie príp. inej funkcie.
- zistí sa početnosť výskytu farieb v oblasti a vyberie sa tá farba, ktorá sa vyskytuje najčastejšie. Ak je výskyt farieb rovnaký, vyberie sa farba podľa iného pravidla alebo ľubovoľná z vyskytujúcich sa farieb.
- vyberie sa farba, ktorá sa vyskytuje najmenej-krát.
- vyberie sa farba ľavého horného bodu oblasti

FILTROVACIE A ROZPTYĽOVACIE METÓDY

SPRIEMERNENIE

4-susedstvo



(11,5,15,7,1)

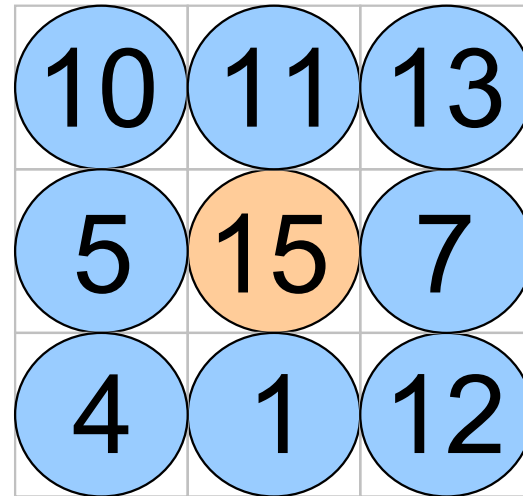


$(11+5+15+7+1=39)/5$



8

8-susedstvo



(10,11,13,5,15,7,4,1,12)



$(10+11+13+5+15+7+4+1+12=78)/9$

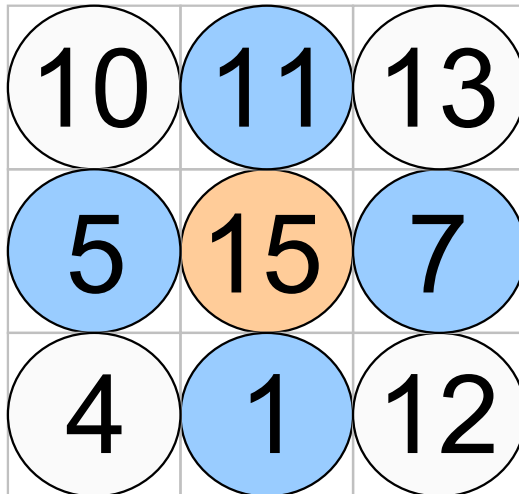


9

FILTROVACIE A ROZPTYĽOVACIE METÓDY

MEDIÁNOVÁ FUNKCIA (MEDIÁN FILTER)

4-susedstvo



(11, 5, 15, 7, 1)

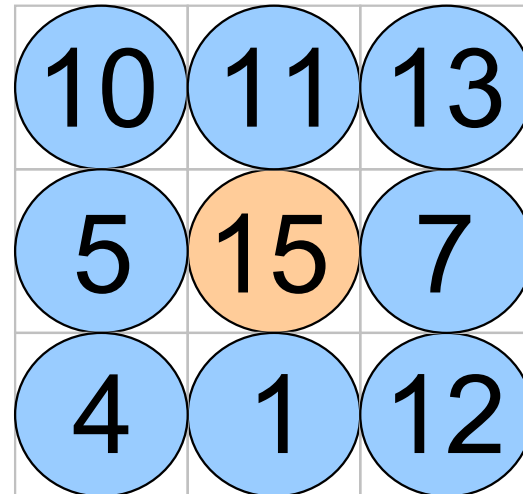


(1, 5, 7, 11, 15)



7

8-susedstvo



(10, 11, 13, 5, 15, 7, 4, 1, 12)



(1, 4, 5, 7, 10, 11, 12, 13, 15)



10

ROZPTYŤOVACIE METÓDY

Metódy na riešenie problému rozdielu medzi požadovanou kvalitou výsledných obrazov a obmedzenými možnosťami dostupného farebného priestoru zariadenia alebo techniky pre podporu *antialiasingu*



ROZPTYŤOVACIE METÓDY

Rozptyľovacie metódy využívajú tú vlastnosť ľudského oka, že z farieb niekoľkých blízkych bodov vytvára dojem jediného bodu, ktorého farba je daná aditívnym zlúčením farieb pôvodných bodov (napríklad pre prevod obrazu so 16 úrovňami šedej na binárny obraz sa použije intenzita jasu (farba) vstupného bodu $I_{IN} \in \langle 0, 15 \rangle$ a výstupného bodu $I_{OUT} \in \langle 0, 1 \rangle$).

- Náhodné rozptyľovanie
- Maticové rozptyľovanie
- Rozptyľovanie distribúciou chyby

0	0	0	0	0	0	0	1	1	1
0	0	0	1	1	1	1	1	1	1
$I_{IN} = 0$		$I_{IN} = 1$		$I_{IN} = 2$		$I_{IN} = 3$		$I_{IN} = 4$	

Príklady rozptyľovacích matic $I_{IN} = \langle 0, 4 \rangle$

0	11	3	14
8	4	10	7
2	13	1	2
9	6	8	5

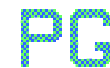
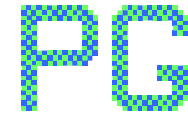
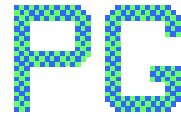
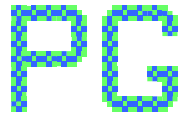
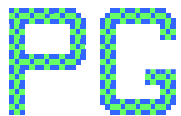
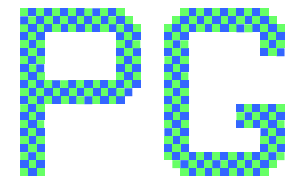
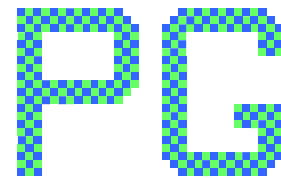
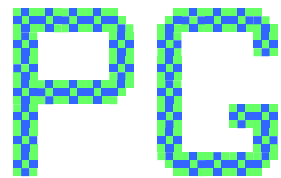
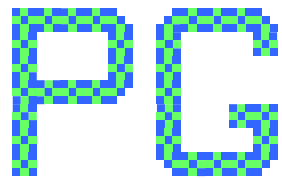
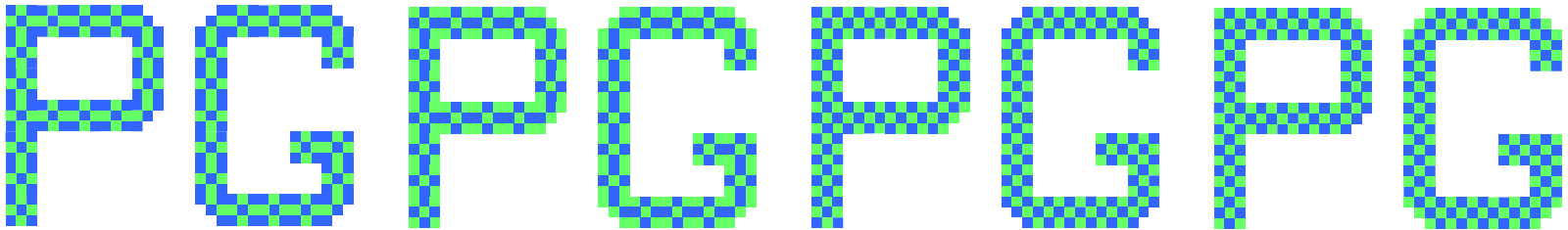
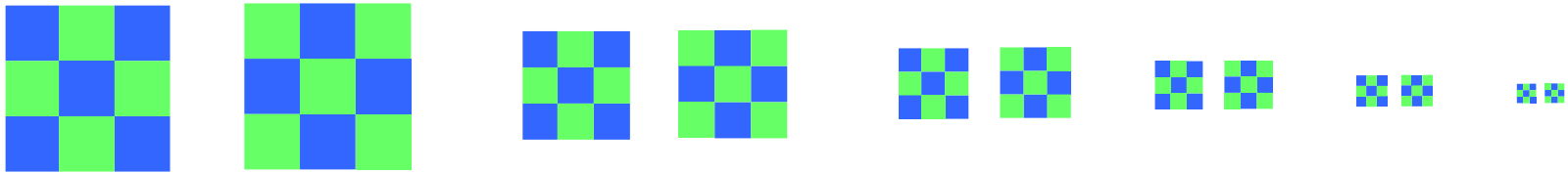
$I_{IN} = \langle 0, 15 \rangle$

0	12	3	15
8	4	11	7
2	14	1	13
10	6	9	5

$I_{IN} = \langle 0, 16 \rangle$

ROZPTYL'OVACIE METÓDY

DITHERING (DVE A VIAC FARIEB)



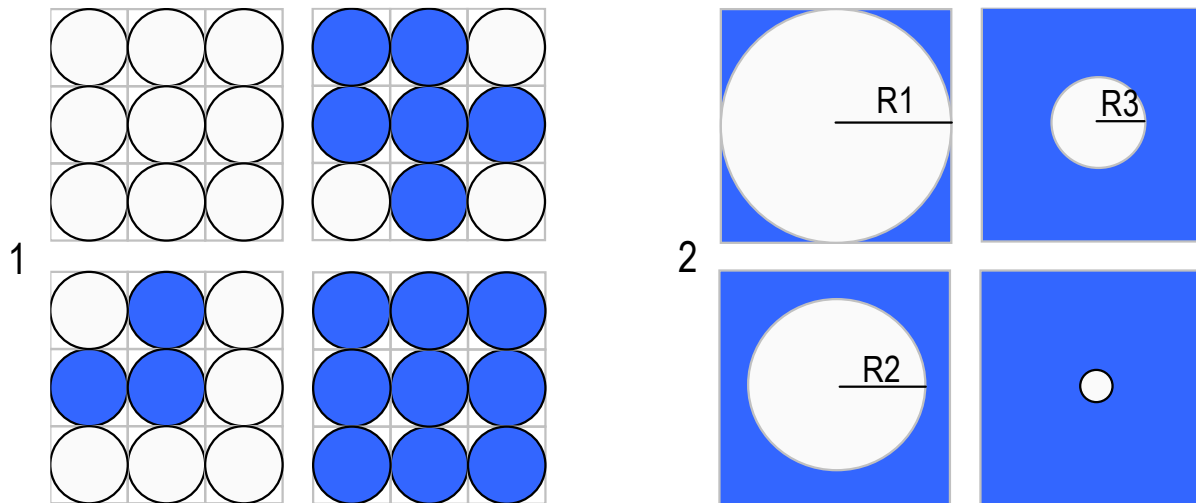
ROZPTYĽOVACIE METÓDY

POLTÓNOVANIE - HALFTONING

Viacero spôsobov ako konvertovať na dve farby. Dôležité je určiť binarizačný prah (threshold).

Najpoužívanejšie spôsoby:

1. *maticové* - využitím binárnych nahradzovacích matíc pre spracovávanú oblasť
2. *funkcionálne* – využitím funkcií k pomerovému vyplneniu spracovávanej oblasti

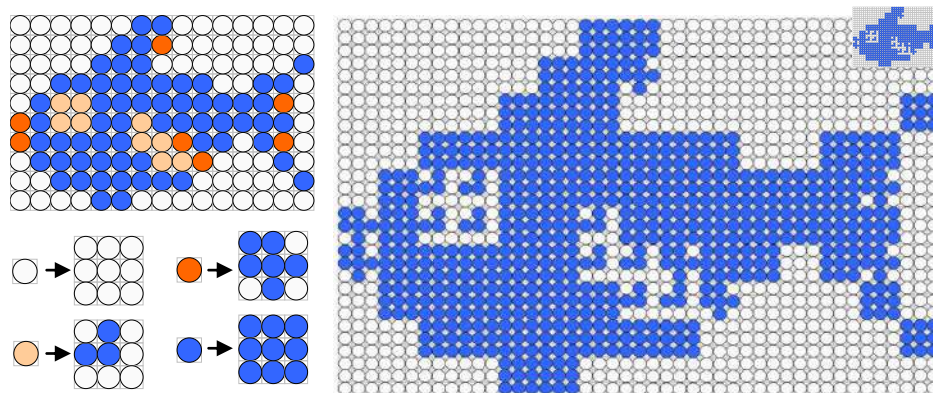
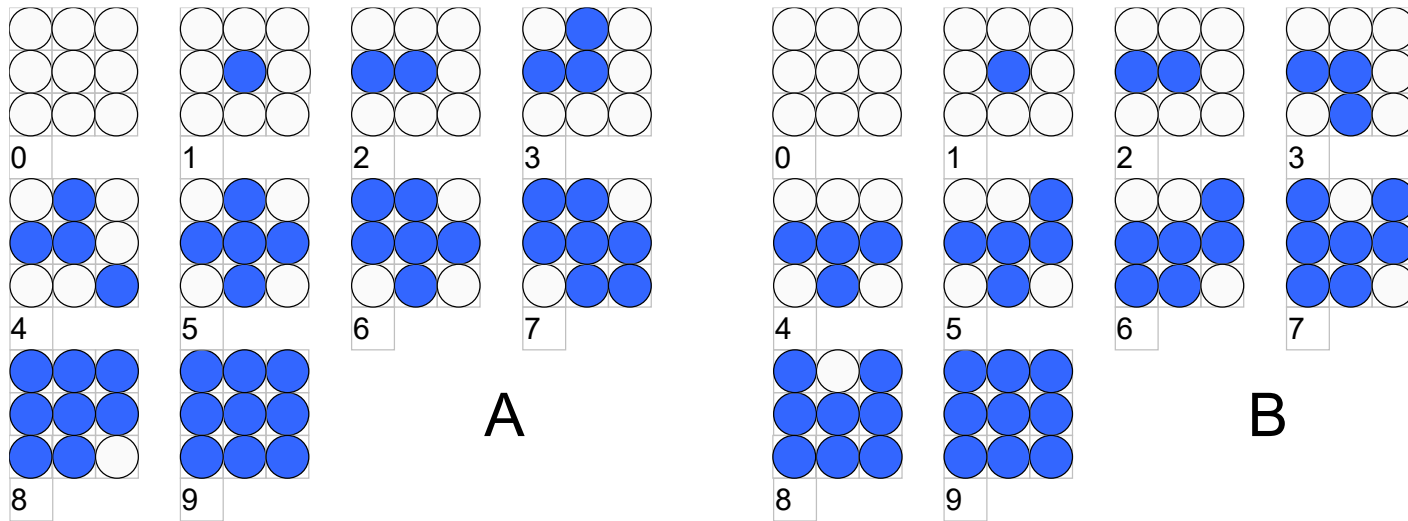


Príklady rôznych spôsobov poltónovania

ROZPTYĽOVACIE METÓDY

POLTÓNOVANIE - HALFTONING

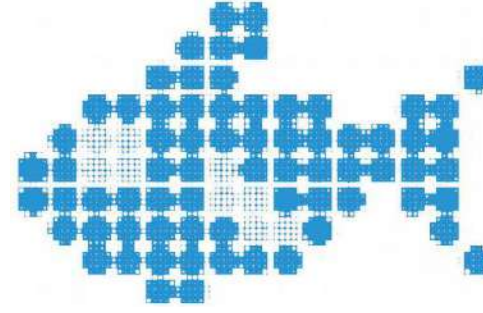
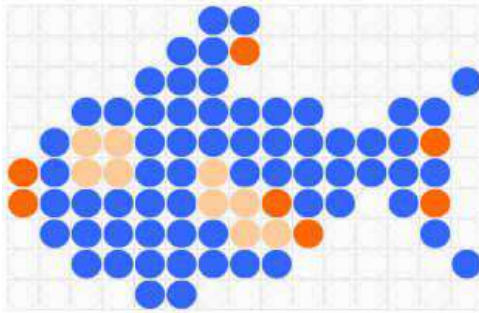
- Rôzne nahradzovacie matice:



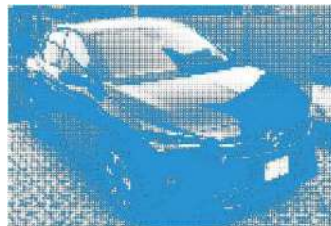
ROZPTYL'OVACIE METÓDY

POLTÓNOVANIE - HALFTONING

- Príklady funkcionálneho poltónovania (rôzne parametre):



a) →



ROZPTYŤOVACIE METÓDY

NA ZÁKLADE DISTRIBÚCIE CHYBY

1. distribúcia chyby - Floyd-Steinberg (koef./16)
2. distribúcia chyby – Stucki (koef./42)
3. distribúcia chyby – Burkes (koef./32)
4. distribúcia chyby – Sierra (koef./32)
5. distribúcia chyby - Jarvis, Judice, Ninke (koef./48)
6. distribúcia chyby - Stevenson, Arce (koef./200)
7. ...

	x	7
3	5	1

1

		x	8	4
2	4	8	4	2
1	2	4	2	1

2

		x	8	4
2	4	8	4	2

3

		x	5	3
2	4	5	4	2
	2	3	2	

4

		x	7	5
3	5	7	5	3
1	3	5	3	1

5

ROZPTYĽOVACIE METÓDY

DISTRIBÚCIA CHYBY - FLOYD-STEINBERG (KOEf./16)

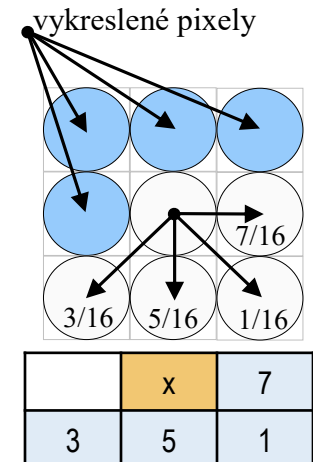
Je to veľmi rýchla metóda, pretože pri nej dochádza k deleniu jednotlivých koeficientov šestnástimi, čo je možné veľmi jednoducho implementovať bitovou rotáciou.

Intenzita vstupného prvku obrazu (bodu) je na výstupe postupne nahradzovaná jednotkou alebo nulou, Vstupná intenzita I_{IN} je použitá nielen k nájdeniu najbližšej zodpovedajúcej intenzity I_{OUT} , ale zanedbaná hodnota, vznikajúca zaokrúhlením I_{IN} z väčšieho rozsahu do rozsahu $<0, 1>$ je využitá na modifikáciu hodnôt susedných prvkov obrazu (Ak má byť napríklad spracovaný prvok obrazu so vstupnou hodnotou 4, výstupnému prvku obrazu sa priradí hodnota 0 a zanedbaná hodnota 4 sa pripočíta k hodnotám susedných prvkov obrazu. Distribuovaná chyba je rozdelená medzi susedné prvky obrazu v závislosti od použitej metódy).

Pred vykreslením každého prvku obrazu je ku vstupnej hodnote I_{IN} pripočítaná doteraz získaná chyba (z pamäte chyby) a až potom je výsledná hodnota prevedená do výstupného rozsahu. Zanedbaná hodnota sa opäť rozdelí a pripočíta do pamäte chyby (nutné zabezpečiť, aby nedochádzalo k strate celočíselným delením).

Postup pre použitie rozptyľovacích metód pre farebné obrazy je nasledujúci:

1. Pre daný obraz sa vytvorí paleta podľa požiadaviek niektorou z metód popísaných vyššie a algoritmus sa nastaví na prvý bod obrazu.
2. Vyberie sa bod a nájde sa k nemu najbližšia farebná hodnota vo vytvorenej palete.
3. Pre daný bod sa určí zanedbaná hodnota, ktorá vznikne zaokrúhlením pôvodnej farby na farbu z palety.
4. Na základe zanedbanej hodnoty a použitej metódy sa rozhodne o prípadnom zvýšení výslednej intenzity farby výsledného bodu.
5. Ak ešte nie je spracovaný celý obraz, posunie sa na ďalší bod a pokračuje sa ďalej krokom 2.



Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

GRAFICKÝ PRIESTOR A OBJEKTY

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 03

© 2025

DIMENZIA (ROZMER) PRIESTORU

Dimenzia je založená spravidla na *fyzikálnej veličine* (jednotke, primárne merateľnej) reprezentujúcu daný priestor alebo jeho zložku. Najčastejšie sa používajú základné a prípadne doplnkové alebo odvodené fyzikálne jednotky (dĺžka=meter, čas=sekunda a pod.). Koncept dimenzie nie je obmedzený na fyzické priestory. Vo fyzike a matematike je rozmer matematického priestoru (alebo objektu) neformálne definovaný ako minimálny počet parametrov (koordinát, súradníc) potrebných na špecifikáciu akéhokoľvek bodu v ňom. Vysokorozmerné priestory často vyskytujúce sa v matematike a prírodných vedách môžu byť priestory parametrov alebo konfiguračné priestory, ako napríklad v Lagrangovej alebo Hamiltonovej mechanike, sú to abstraktné priestory, nezávislé od fyzického priestoru, v ktorom žijeme.

Typy dimenzií:

- Číselné
- Nečíselné

Druhy dimenzií:

- Topologická
- Hausdorffova
- Farebná
- Fraktálna
- dimenzia sebedobnosti
- Kapacitná
- Informačná
- dimenzia Rotácií
- ...



ŠTRUKTÚRA DIMENZIE (ROZMERU) PRIESTORU

$$D(imenzia) = M + N + \dots$$

veličina 1
veličina 2
veličina n

Príklad štruktúry 3 hodnotovej dimenzie:

$3 = 3 + 0$ (geometria + čas) – *homogénna štruktúra*

$3 = 2 + 1$ (geometria + čas) – *nehomogénna (heterogénna) štruktúra*



VRSTVY VIZUALIZAČNÉHO PROCESU

1. Definovanie/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



PRIESTOR A JEHO PARAMETRE

Podľa charakteru priestoru (definuje aj DOF):

1. Translačný priestor (T)
2. Rotačný priestor (R)
3. Kombinovaný priestor (C)

Podľa typu dimenzie:

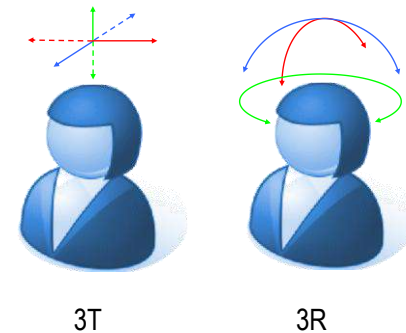
1. Celočíselné – topologické (I)
2. Neceločíselné (F)

Podľa štruktúry dimenzií (primárne geometria) a ich počtu:

1. Homogénna štruktúra ($N+0$) – ND priestor (U_{niform})
2. Heterogénna štruktúra ($N_1+N_2+....N_n$) – $(N_1+N_2+....N_n)D$ priestor (H)

Príklady:

- celočíselný 3D translačný priestor (IU3DT),
- neceločíselný 2+1D rotačný priestor (FH2d1DR)



OBJEKTY V POČÍTAČOVEJ GRAFIKE

1. 0-rozmerný objekt (bod)
 2. 1-rozmerný objekt (priamka, úsečka)
 3. 2-rozmerný objekt (plocha)
 4. 3-rozmerný objekt (teleso)
- Všetky geometrické objekty, sa dajú spojitou transformáciou previesť na vyššie uvedené objekty ak majú rovnakú topologickú dimenziu.
 - Všetky geometrické objekty, môžu reprezentovať vyššie uvedené objekty, ak majú rovnakú alebo nižšiu topologickú dimenziu
 - Počet menených súradníc/parametrov transformácií v danej aktuálnej topologickej dimenzii (ATD) je určený vzťahovými objektami (VZO, napr. rotácia voči VZO(bod, os a pod) je daný *doplnkovo* k ATD. A naopak ak v ATD je počet menených parametrov transformácie N , potom maximálna dimenzia VZO bude doplnok k ATD (t.j. $ATD - N$)
 - Napríklad ak sa bude v 3D priestore zrkadliť voči osi (1-rozmerný objekt) počet menených parametrov bude $3 - 1 = 2$. Naopak, ak bude v 2D priestore požadovaná rotácia (mení 2 parametre) potom táto je možná len okolo počiatku sústavy (0-rozmerný objekt, $2 - 2 = 0$).

POPIS A REPREZENTÁCIA OBJEKTOV

Pri spracovaní objektov sú zaujímavé dve hľadiská:

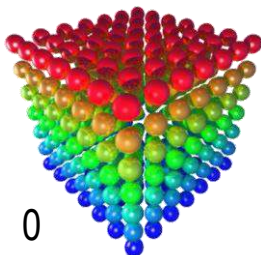
- popis objektu
- reprezentácia objektu

Pre modelovanie telies sú používané tri základné spôsoby popisu:

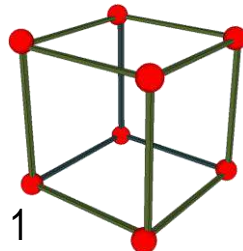
- hraničná reprezentácia a jej štruktúrovaná derivácia B-rep
- konštruktívna geometria telies (CSG)
- vypočítavanie obsadených častí priestoru

POPIS A REPREZENTÁCIA OBJEKTOV

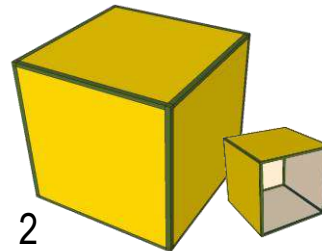
- systémy založené na množine bodov – mračná bodov (Points clouds) • 0
- systémy založené na drôtovom modeli (Wire Frame Model) • 1
- systémy založené na povrchovom modeli (Surface Model) • 2
- systémy založené na objemovom modeli (Solid Model) • 3



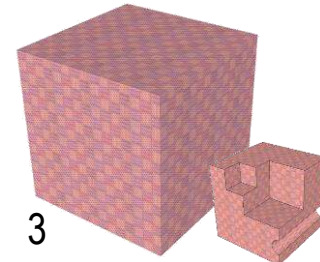
0



1



2



3

HRANIČNÁ REPREZENTÁCIA

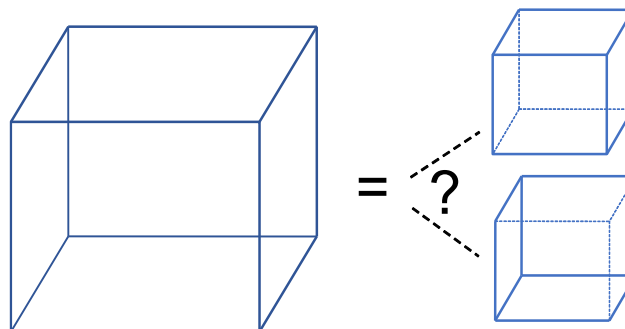
Vychádza z predstavy, že najvýznamnejšou časťou telesa sú jeho hraničné elementy ako napr. hrany alebo povrch, ktorý tvorí hranicu medzi hmotou telesa a okolitým priestorom.

Hraničné plochy môžeme deliť na časti rovín, analytické plochy a na špeciálne parametrické plochy.

Najjednoduchšia metóda popisu hranice telies spočíva v stanovení hrán a vrcholov na povrchu telesa (drôťový model).

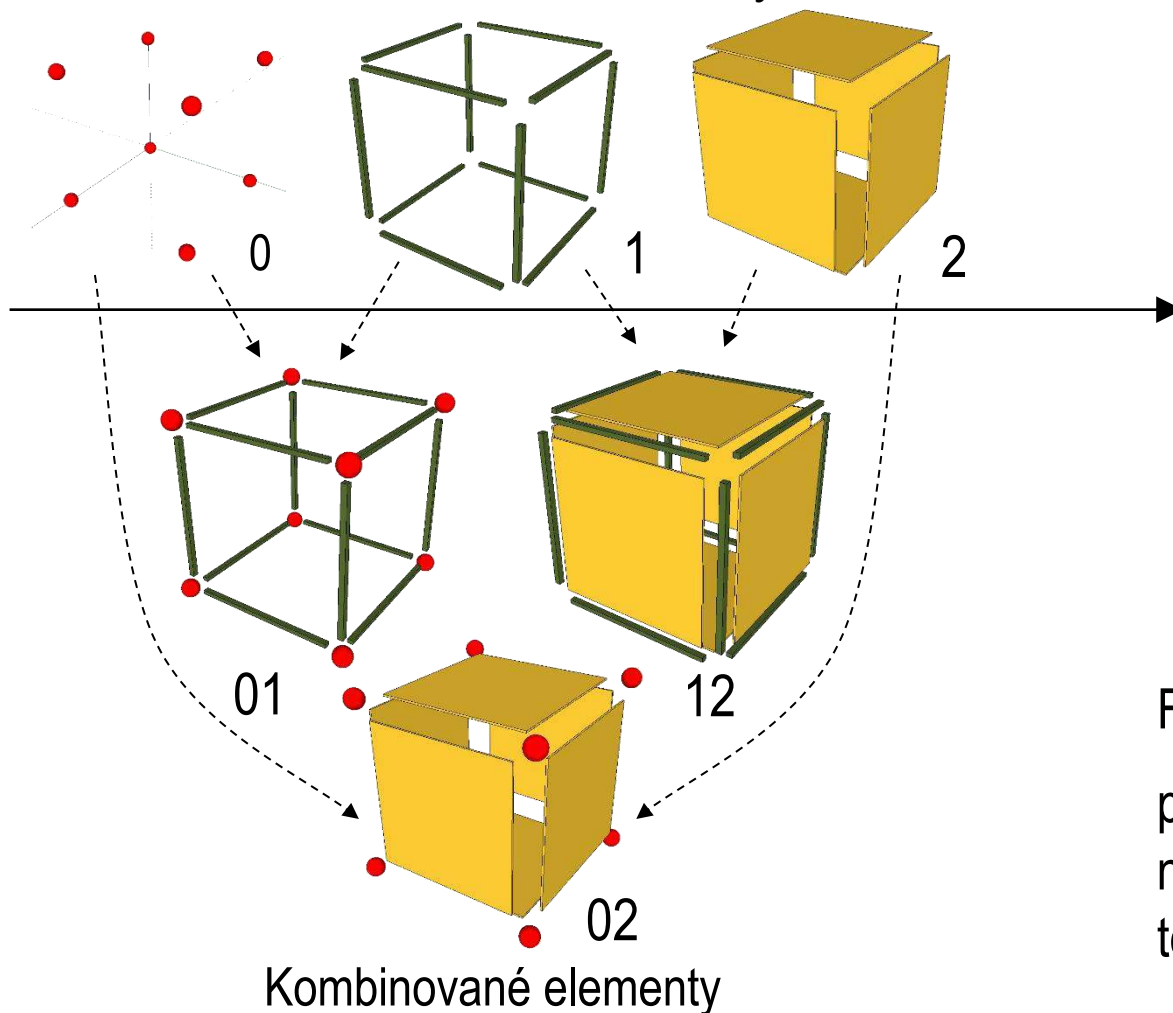
Môže byť nejednoznačná

(spravidla ak sa n -rozmerný objekt spracováva v $n-1$ a menej dimenzionálnom priestore)



HRANIČNÁ REPREZENTÁCIA

základné elementy



Formát:

podľa implementačnej potreby,
najčastejšie binárny alebo
textový

HRANIČNÁ REPREZENTÁCIA

ŠTRUKTUROVANÁ B-REP METÓDA

B-REP (Boundary Representation) definuje objekt svojím povrchom.

Povrch je zložený zo stien, ktoré sa môžu vzájomne dotýkať iba na spoločných hranách, pričom každá hrana je orientovaná (dá sa jednoznačne určiť, či sa jedná o vnútornú, alebo vonkajšiu hranu).

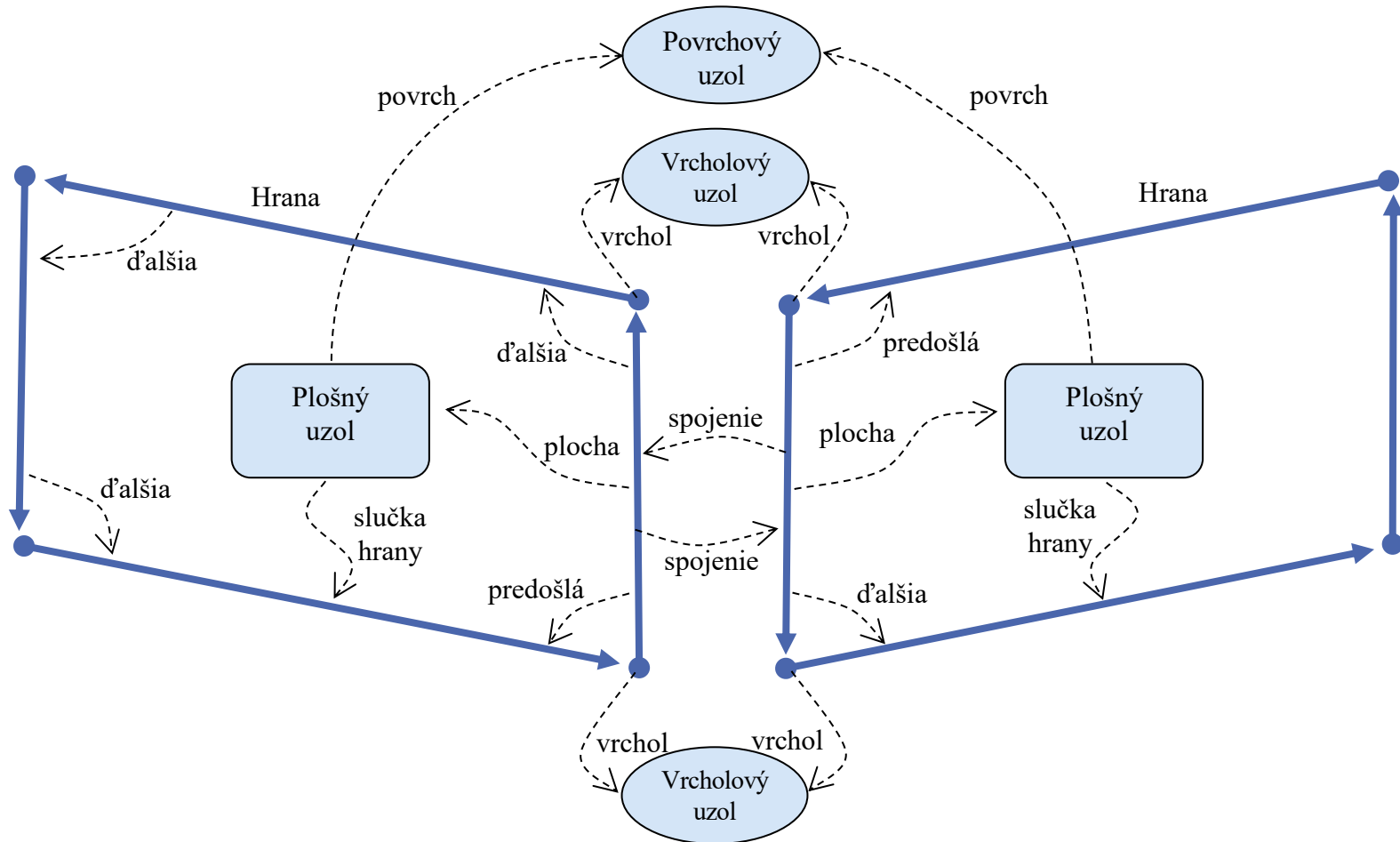
Objekt je reprezentovaný dynamickou údajovou štruktúrou nazývanou **Winged Edge Structure**, zloženou zo štyroch druhov uzlov:

- *vrchol (Vertex)*,
- *hrana (Edge)*,
- *stena, plocha (Face)*,
- *teleso (Solid)*



HRANIČNÁ REPREZENTÁCIA

ŠTRUKTUROVANÁ B-REP METÓDA



Rozšírená hranová (winged-edge) štruktúra

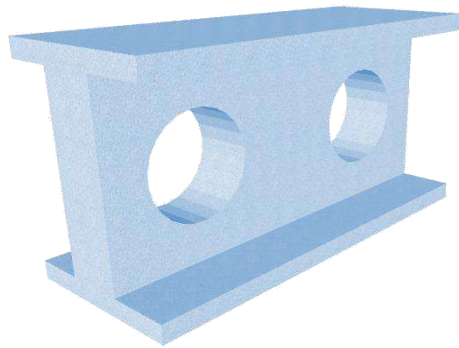
KONŠTRUKTÍVNA GEOMETRIA TELIES

CSG – CONSTRUCTIVE SOLID GEOMETRY

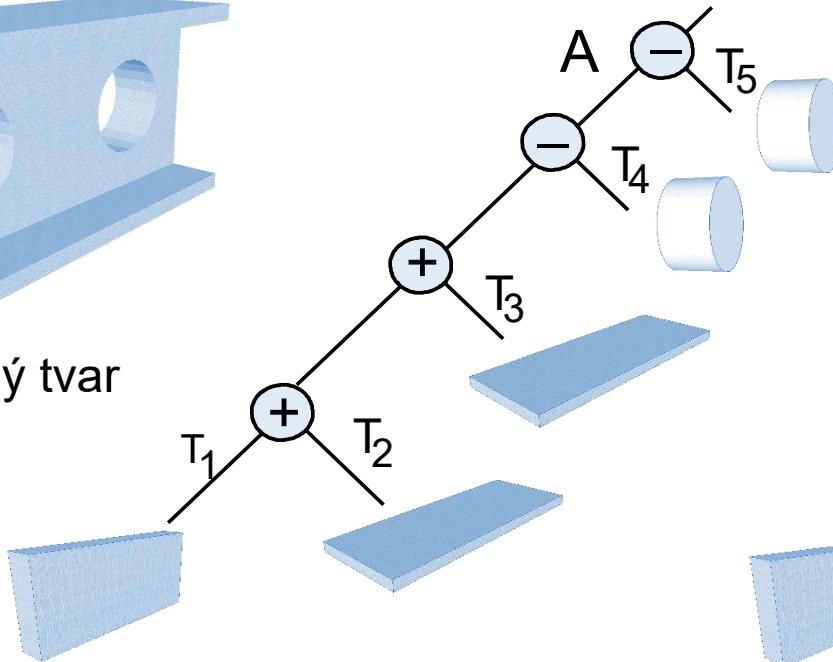
Je definovaná abstraktnou údajovou štruktúrou: **strom**

- **Listy** stromu definujú jednotlivé atomárne elementy, z ktorých sa následne objekt skladá
- **Uzly** stromu definujú operácie medzi atomárnymi elementami/objektami a/alebo prípadne vzniknutými subobjektami na danej úrovni stromu (zjednotenie, rozdiel, prienik)
- **Hrany** stromu definujú transformácie atomárnych elementov a/alebo subobjektov vstupujúcich do rodičovského uzla
- v **Koreni** stromu je už definovaný celý objekt

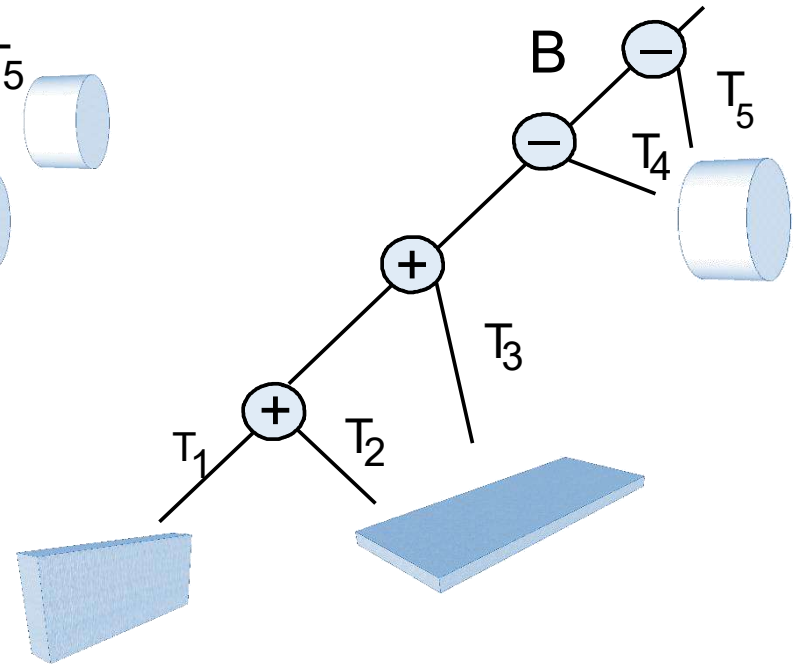
CSG REPREZENTÁCIA



výsledný tvar

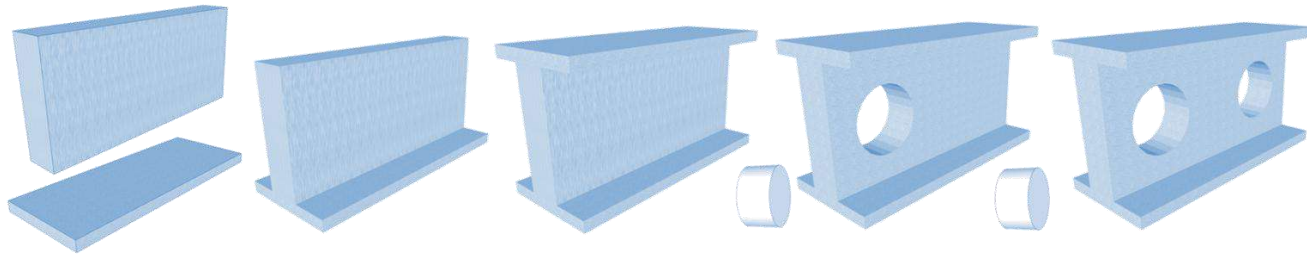


acyklický strom

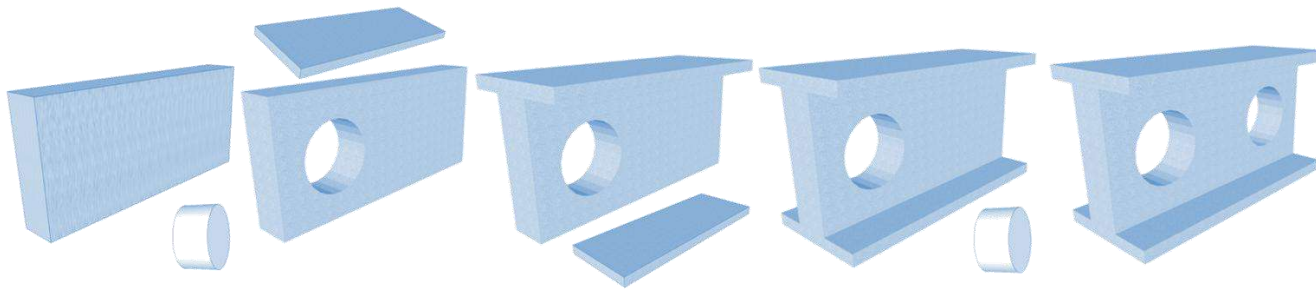
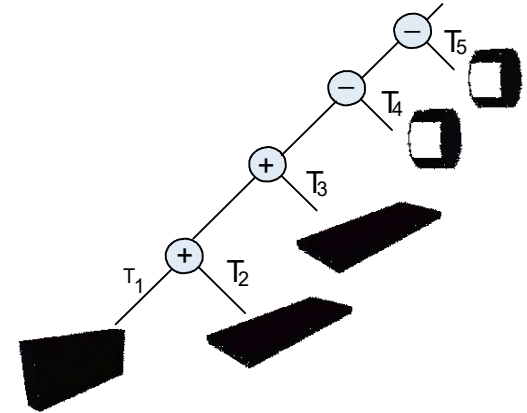


cyklický strom

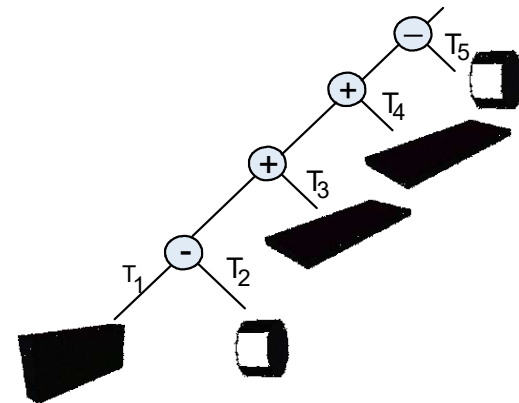
CSG REPRESENTÁCIA



A



B



PRIESTOR A JEHO SÚRADNICOVÁ SÚSTAVA

KOORDINAČNÝ SYSTÉM

- *Súradnicová sústava* (koordinačný systém) umožňuje parametrizovať priestor a definovať jeho počiatočný bod (stred súradnicovej sústavy, origin) a smery rozvoja fyzikálnej veličiny v príslušnej dimenzii priestoru popisovaného koordinačným systémom (jeho osi).
- *Súradnice* (parametre, koordináty) definujú jednoznačne polohu v rámci súradnicovej sústavy podľa charakteru a štruktúry dimenzií priestoru



SÚRADNICOVÉ SÚSTAVY

PARAMETRE

Súradnicová sústava umožňuje parametrizovať priestor a definovať jeho počiatočný bod

1. *Počiatok* (stred súradnicovej sústavy)
2. *Os* – definuje smer rozvoja fyzikálnej veličiny v príslušnej dimenzii priestoru popisovaného súradnicovou sústavou
3. *Súradnice* (parametre) – definujú jednoznačne polohu v rámci súradnicovej sústavy podľa charakteru a štruktúry dimenzií priestoru (Počet súradníc – podľa počtu dimenzií priestoru)
4. *Smer rozvoja zoradenia súradníc* – najčastejšie točivosť (ľavo alebo pravotočivá)

SÚRADNICOVÉ SÚSTAVY

LINEARITA A PRAVOUHLOŠŤ

Podľa rozvoja hodnôt veličín príslušných dimenzií priestoru ktorému je súradnicová sústava priradená:

1. Lineárne súradnicové sústavy
2. Nelineárne súradnicové sústavy

Podľa vzťahu osí súradnicového systému:

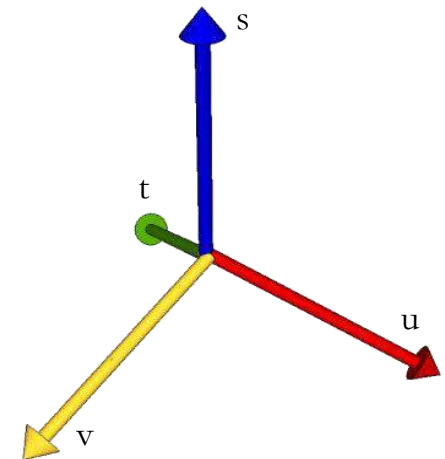
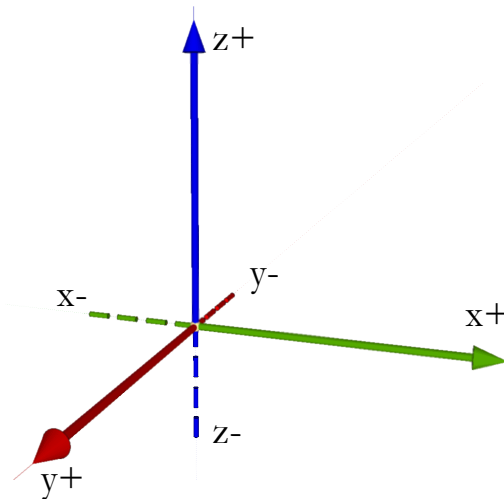
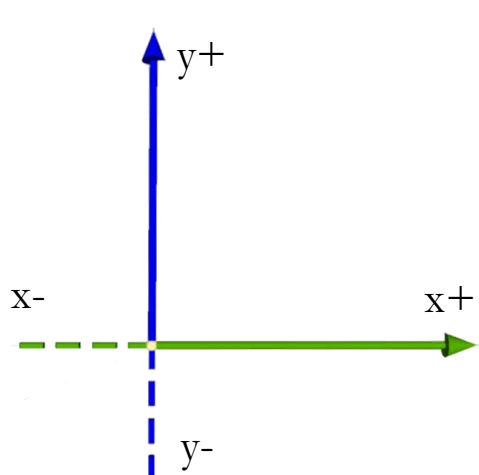
1. Pravouhlé
2. Nepravouhlé

Pre počítačovú grafiku sa primárne používajú celočíselné translačné alebo rotačné lineárne súradnicové sústavy

SÚRADNICOVÉ SÚSTAVY

NAJPOUŽÍVANEJŠIE TYPY

1. 2D (topologicky, 2+0) translačná pravouhlá lineárna súradnicová sústava
2. 3D (topologicky, 3+0) translačná pravouhlá lineárna súradnicová sústava
3. 4D (topologicky, 4+0) translačná pravouhlá lineárna súradnicová sústava (projekcia do 3D, disfenoid)

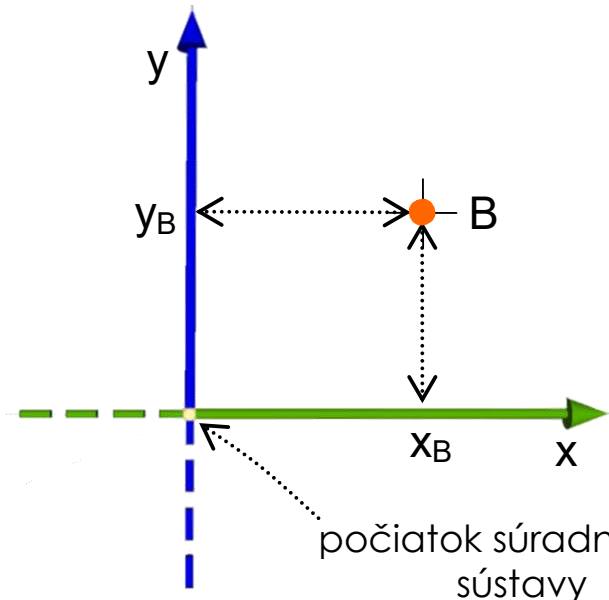


SÚRADNICOVÉ SÚSTAVY

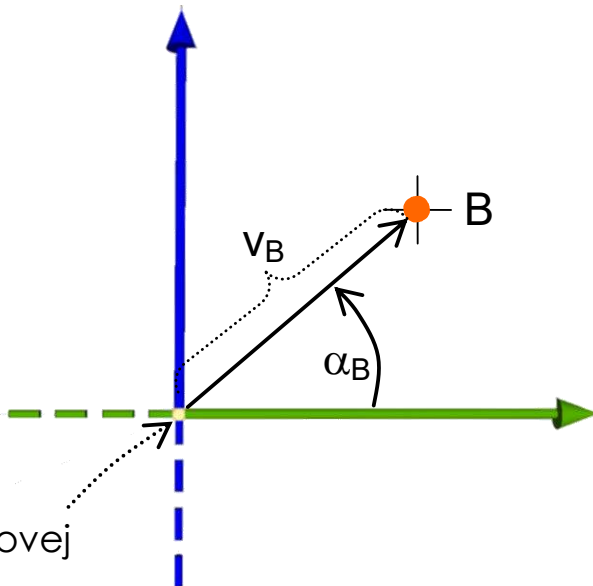
NAJPOUŽÍVANEJŠIE 2D TYPY

1. Karteziánska 2D pravouhlá súradnicová sústava
2. Polárna súradnicová sústava

Pravouhlé karteziánske
súradnice bodu B: $B[x_B, z_B]$



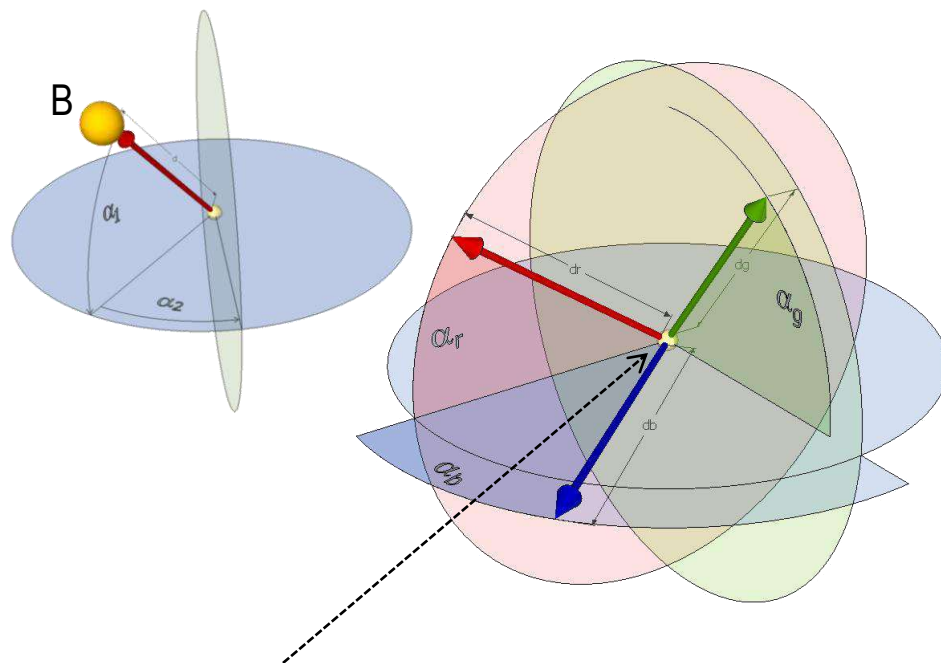
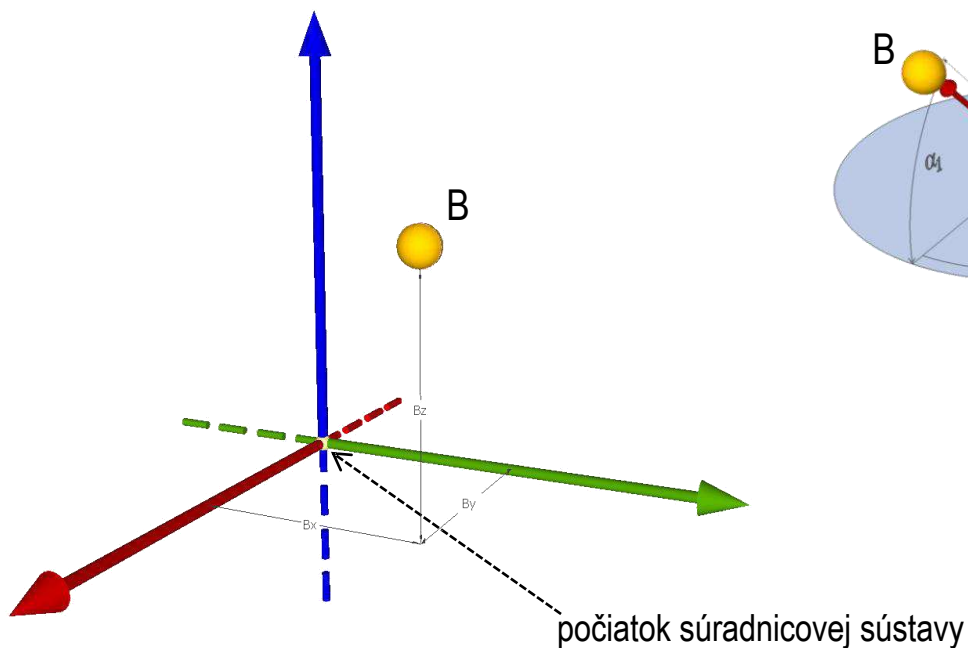
Polárne súradnice bodu B:
 $B[v_B, \alpha_B]$



SÚRADNICOVÉ SÚSTAVY

NAJPOUŽÍVANEJŠIE 3D TYPY

1. Karteziánska 3D pravouhlá súradnicová sústava
2. Sférická súradnicová sústava



SÚRADNICOVÉ SÚSTAVY

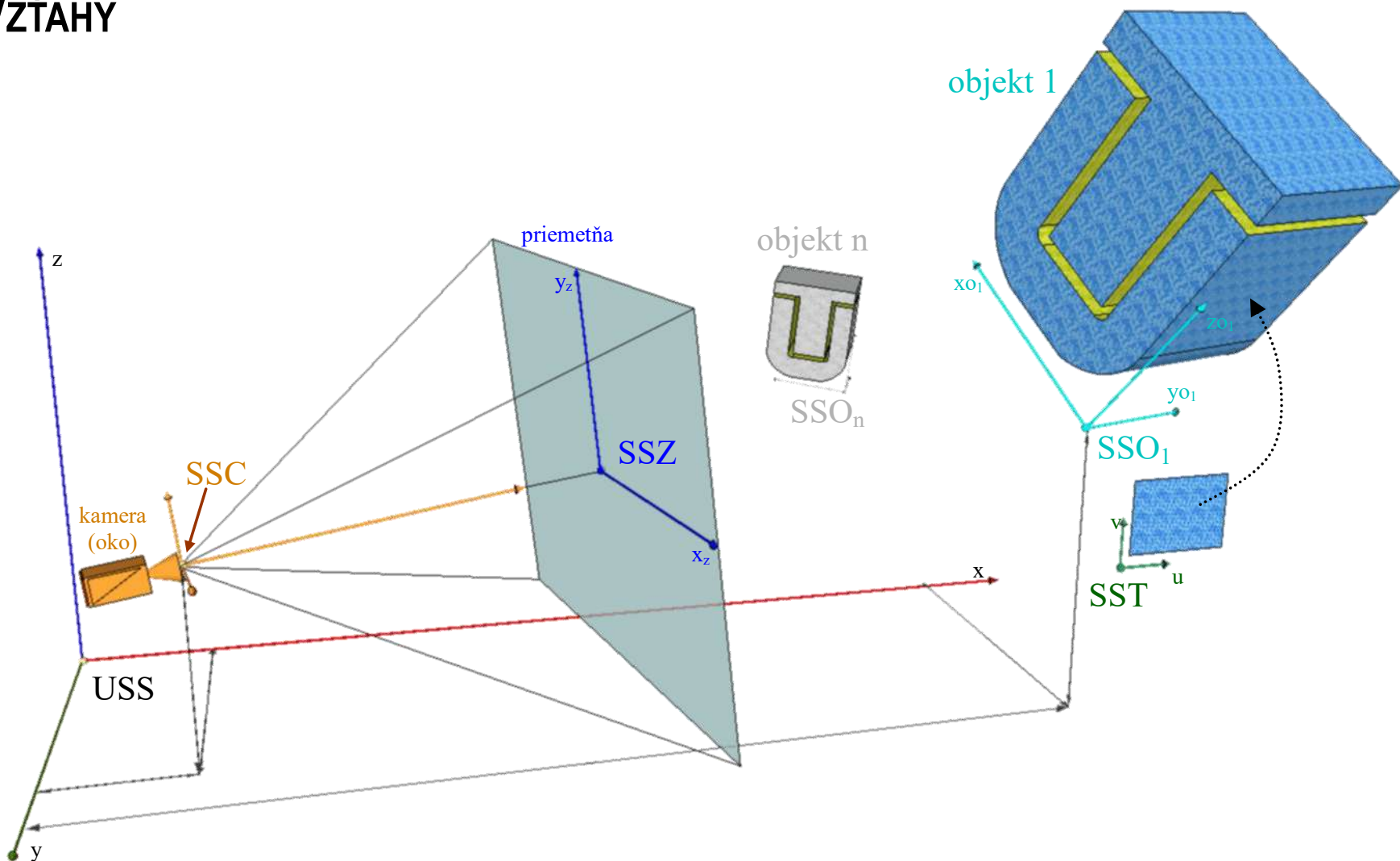
ŠTANDARDNÉ PRACOVNÉ PRIESTORY

1. *USS* - Univerzálna (Používateľská, Globálna) Súradnicová Sústava
2. *SSO* - Súradnicová Sústava Objektu
3. *SST* - Súradnicová Sústava Textúry
4. *NSS* - Normalizovaná Súradnicová Sústava
5. *SSC* - Súradnicová Sústava kamery
6. *SSC₂* - dvojkamerová stereoskopická Súradnicová Sústava
7. *SSZ* - Súradnicová Sústava Zariadenia (logická
fyzická)



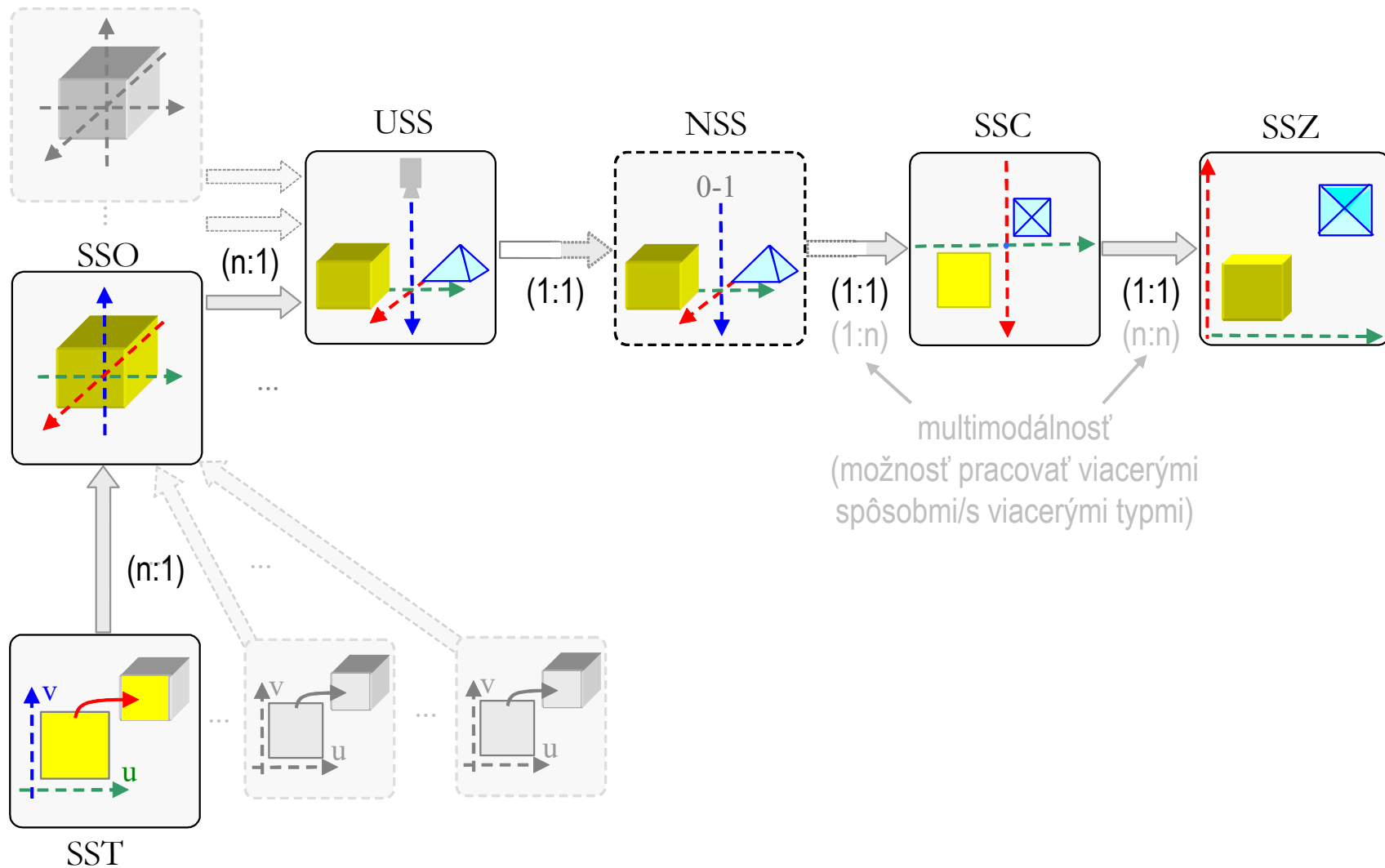
SÚRADNICOVÉ SÚSTAVY

VZŤAHY



SÚRADNICOVÉ SÚSTAVY

VZŤAHY – SÚRADNICOVÝ REŤAZEC



Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

TRANSFORMÁCIE V POČÍTAČOVEJ GRAFIKE

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 04

© 2025

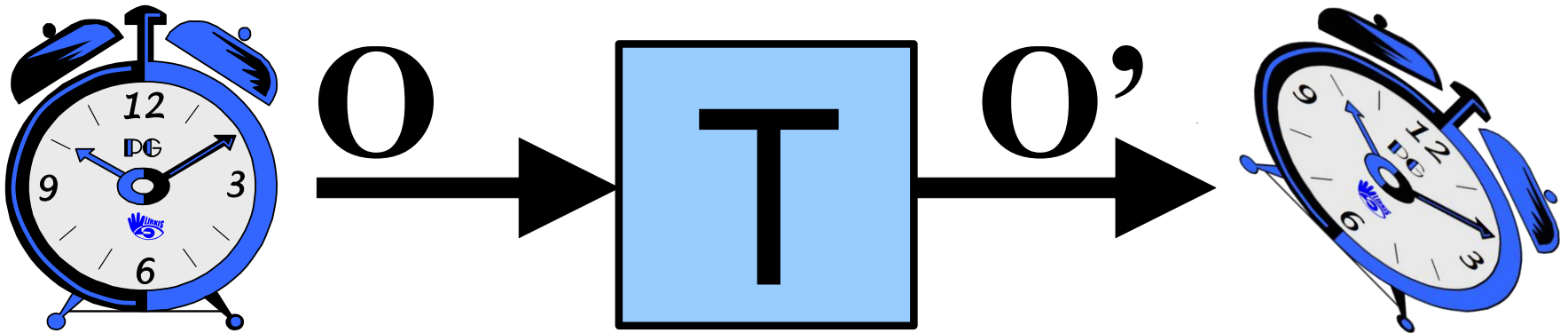
VRSTVY VIZUALIZAČNÉHO PROCESU

1. Definovanie/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



TRANSFORMÁCIE

Transformácia je proces, ktorý mení (transformuje) vstupný objekt (jeho parametre) na objekt výstupný.



TRANSFORMÁCIE

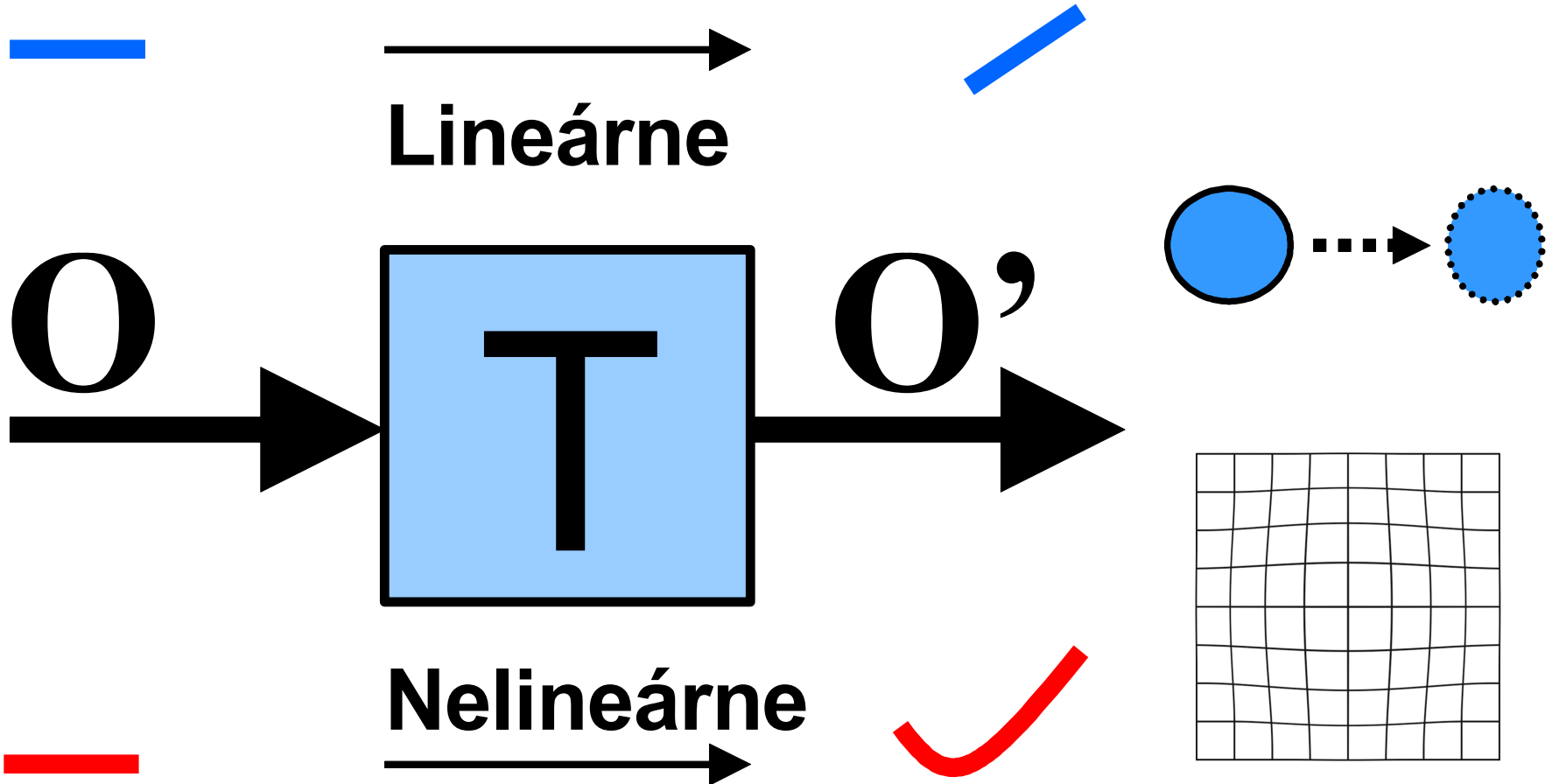
PODĽA SCHOPNOSTI MENIŤ CHARAKTER OBJEKTU

- **Lineárne** – po transformácii sa nemení charakter objektu (počet a typ parametrov/atribútov). Medzi tieto transformácie radíme najmä posunutie (transláciu), otočenie (rotáciu), zmenu mierky (škálovanie), skosenie a zrkadlenie.
- **Nelineárne** – po transformácii sa mení charakter objektu (počet a typ parametrov/atribútov). Medzi tieto transformácie patrí napr. distorzia obrazu, rybie oko, panoráma, zošikmenie, face warp a pod.



TRANSFORMÁCIE

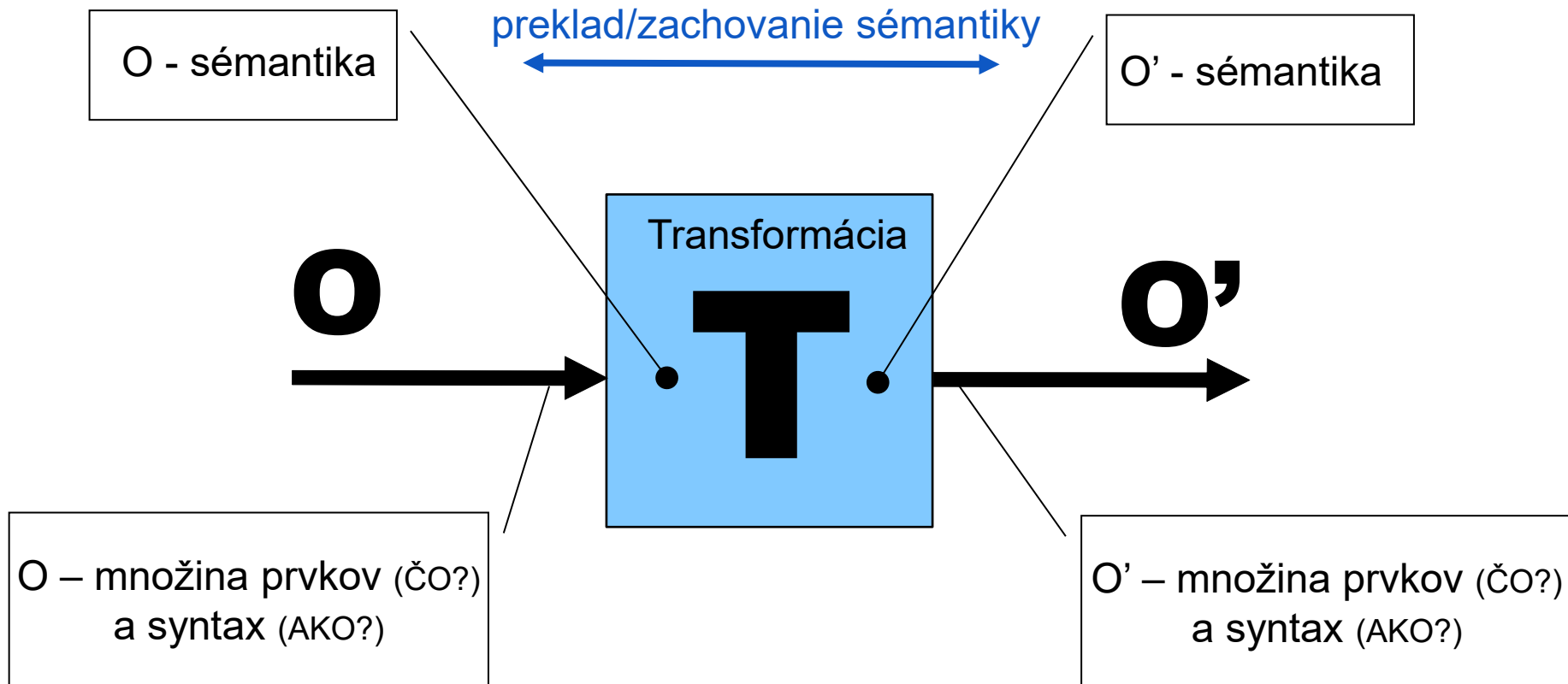
PODĽA SCHOPNOSTI MENIŤ CHARAKTER OBJEKTU



TRANSFORMÁCIE

PODĽA SCHOPNOSTI ZACHOVAŤ SÉMANTIKU

- so zachovaním sémantiky (spravidla lineárne)
- s prekladom sémantiky (spravidla nelineárne)



TRANSFORMÁCIE

PODĽA PRIESTORU, V KTOROM SA APLIKUJÚ

1. *Filtrovanie (filtering)* – Aplikácia v priestore farieb. Napr. ak sa aplikuje na najjednoduchší grafický prvok a to pixel, potom sa spravidla aplikuje na atribút jeho farby. Používa sa najmä v spracovaní obrazu.
2. *Deformovanie (warping)* – Aplikácia v geometrickom priestore. Napr. ak sa aplikuje na najjednoduchší grafický prvok a to pixel, potom sa spravidla aplikuje na atribút jeho polohy. Používa sa najmä u geometrických transformácií v rámci počítačovej grafiky

Vizuálne výsledky v rámci PG sú viditeľné najlepšie pri aplikácii na množine viacerých grafických prvkov napr. u rastrovej grafiky na obraze.



filter

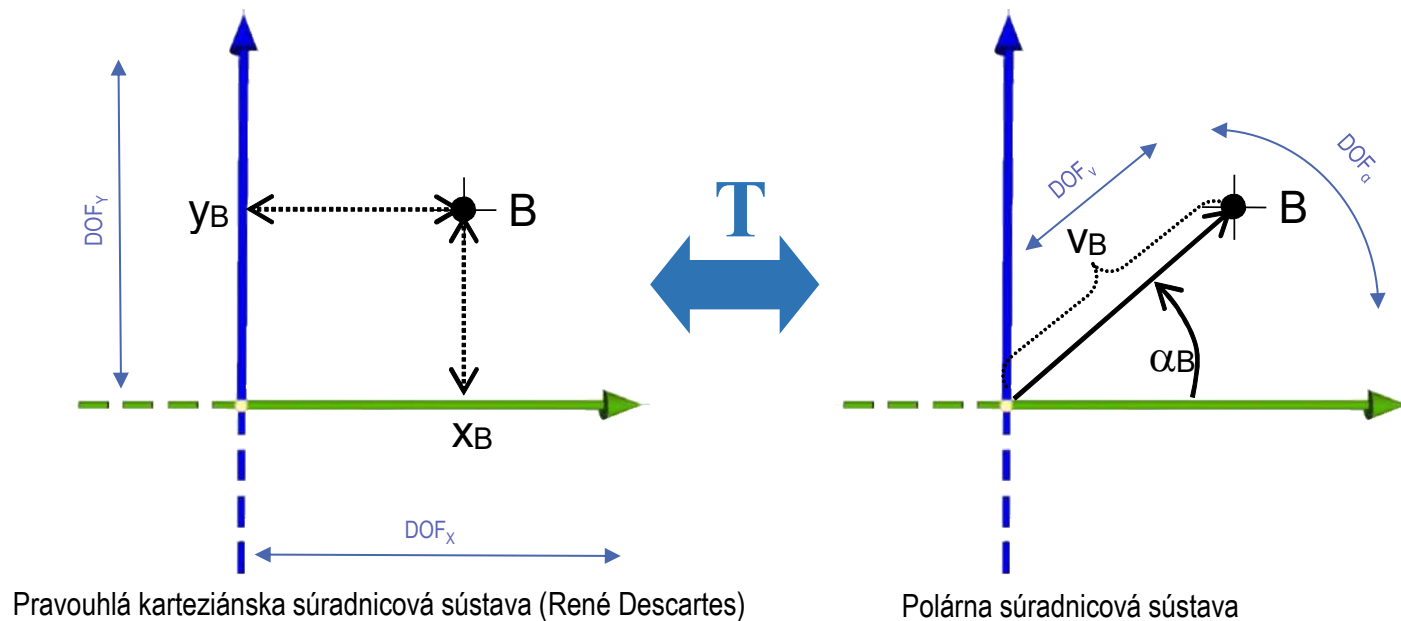


deformácia



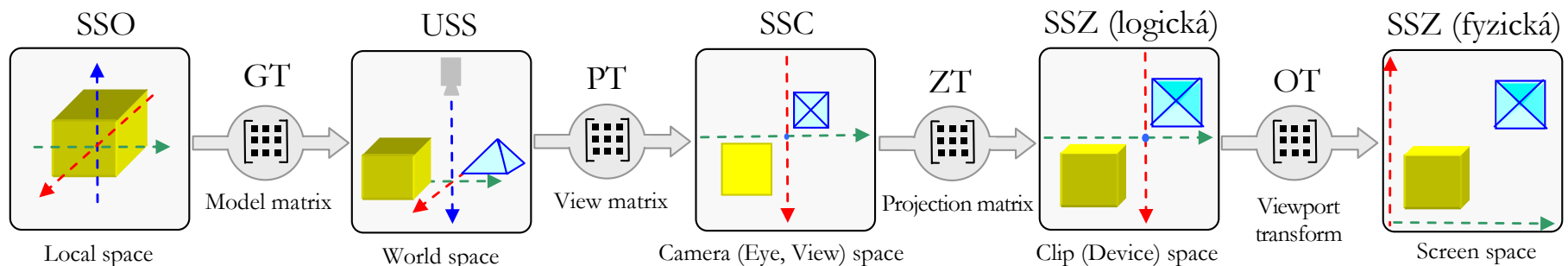
VZŤAH PRIESTOROV, ICH SÚRADNICOVÝCH SÚSTAV A TRANSFORMÁCIÍ

- Z hľadiska použitého typu sústavy je možné príslušnú transformáciu z hľadiska jej vonkajšieho prejavu v rôznych sústavách vnímať rozdielne.
- Rôzne manipulácie toho istého objektu je možné jednoduchšie popísať použitím inej sústavy a vyjadrením transformácie medzi týmito sústavami
- Vo virtuálnom grafickom svete DOF (Degree of Freedom – stupeň voľnosti) definuje počet translačných a rotačných (alebo časových) smerov, v ktorých sa môže grafický objekt pohybovať).



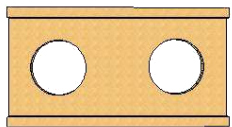
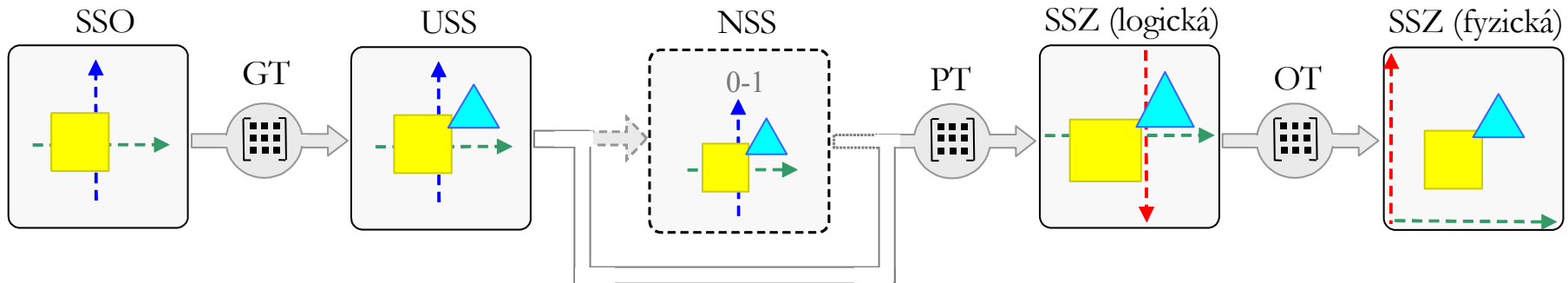
MEDZISÚSTAVOVÉ TRANSFORMÁCIE

- Globálna (geometrická) transformácia **GT**,
- Pohľadová transformácia **PT**,
- Zobrazovacia (projekčná) transformácia **ZT**,
- Orezávací transformácia **OT**



Doplnkovo **Mapovacia transformácia (MT)** pri aplikácii textúry (Texture space, uv space) v rámci vzťahu SST→SSO

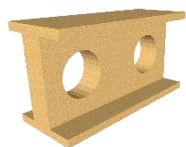
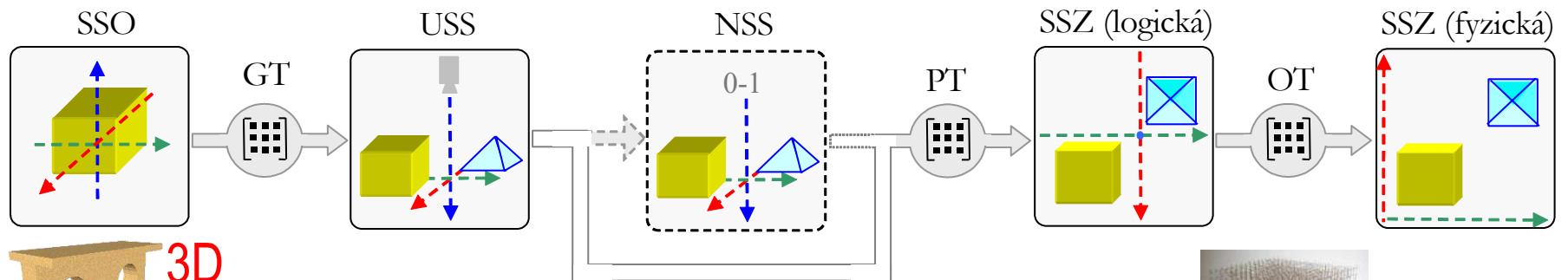
TRANSFORMAČNÉ REŤAZCE ND→ND



2D



2D

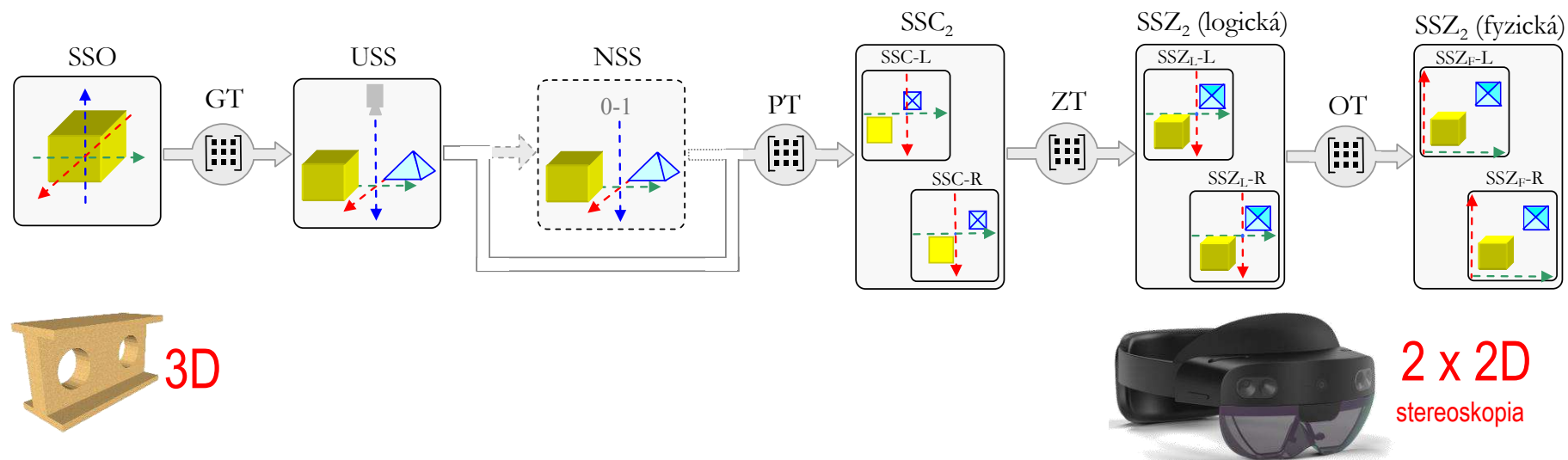
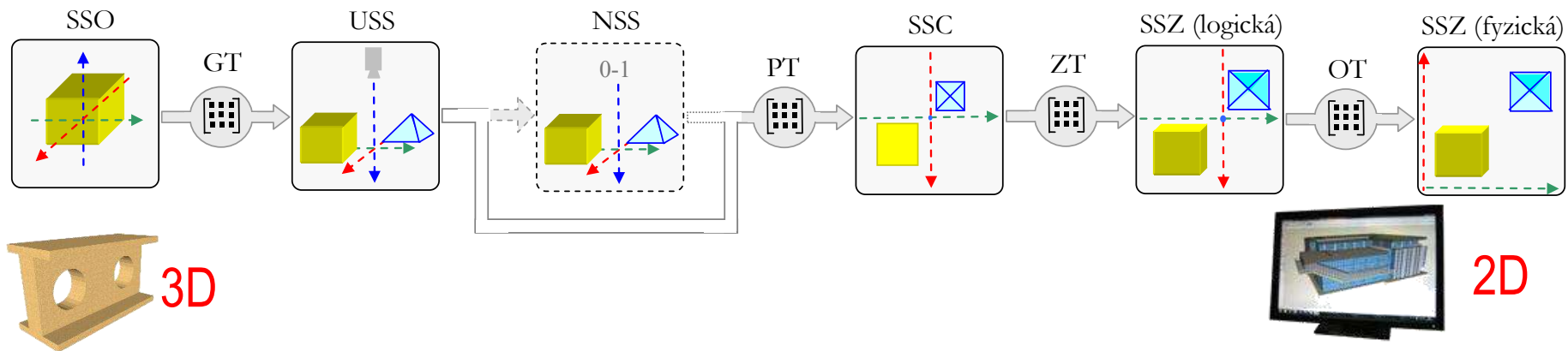


3D

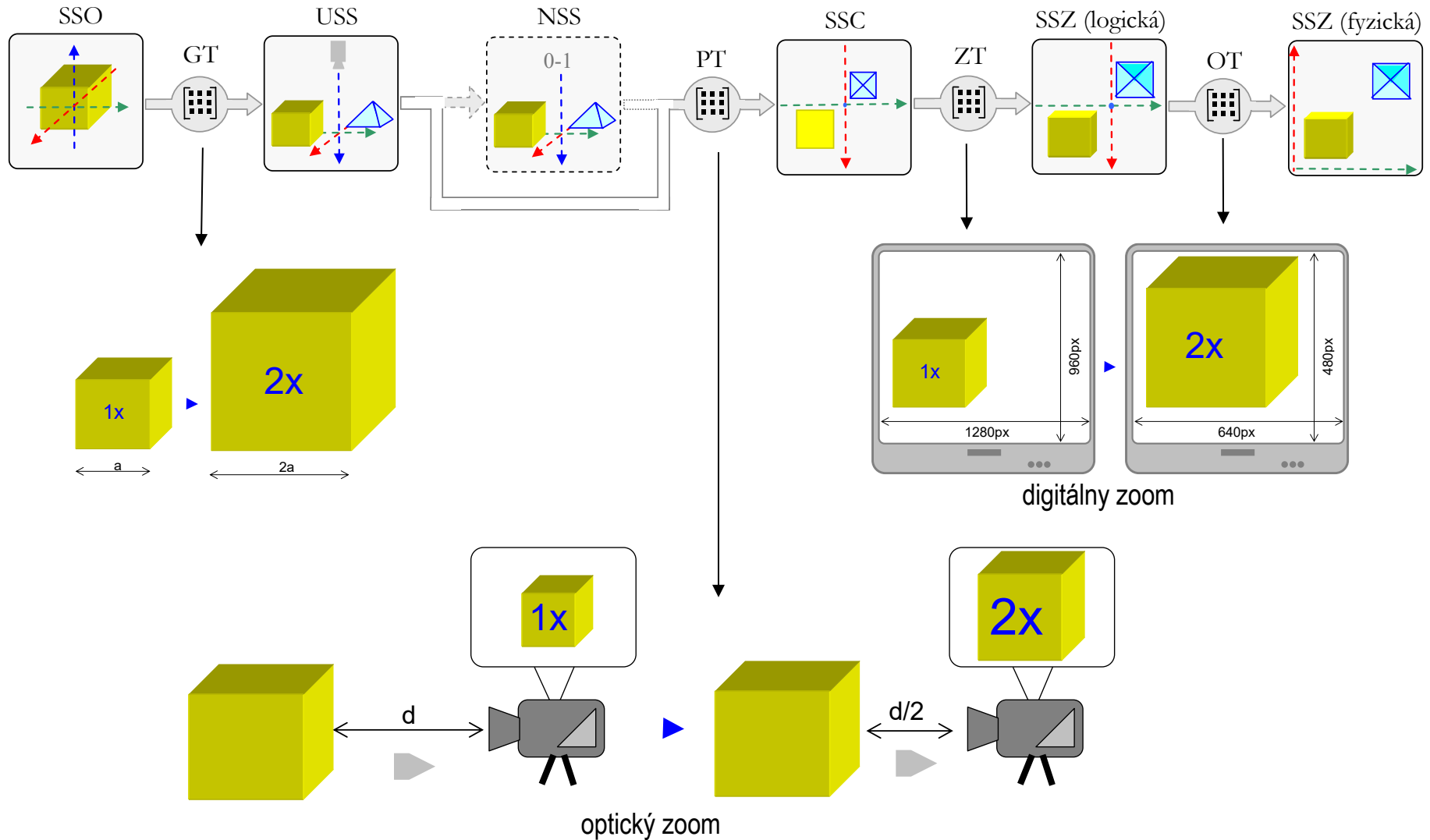


3D

TRANSFORMAČNÉ REŤAZCE $ND \rightarrow (N-1)D$



APLIKÁCIA TRANSFORMÁCIÍ V ZOBRAZOVACOM REŤAZCI



IMPLEMENTÁCIA TRANSFORMÁCIÍ, TRANSFORMAČNÉ MATICE

- Implementácia analytickým spôsobom
- Implementácia pomocou maticového počtu

$$O \times T = O'$$

Transformačná
matica

$$[x, y] * T = [x', y'] \quad \text{v 2D}$$

$$[x, y, z] * T = [x', y', z'] \quad \text{v 3D}$$

Podstatné obmedzenie matic 3x3 (3D): nemožnosť použitia pre posunutie
 Riešenie: zavedenie homogénnych súradníc

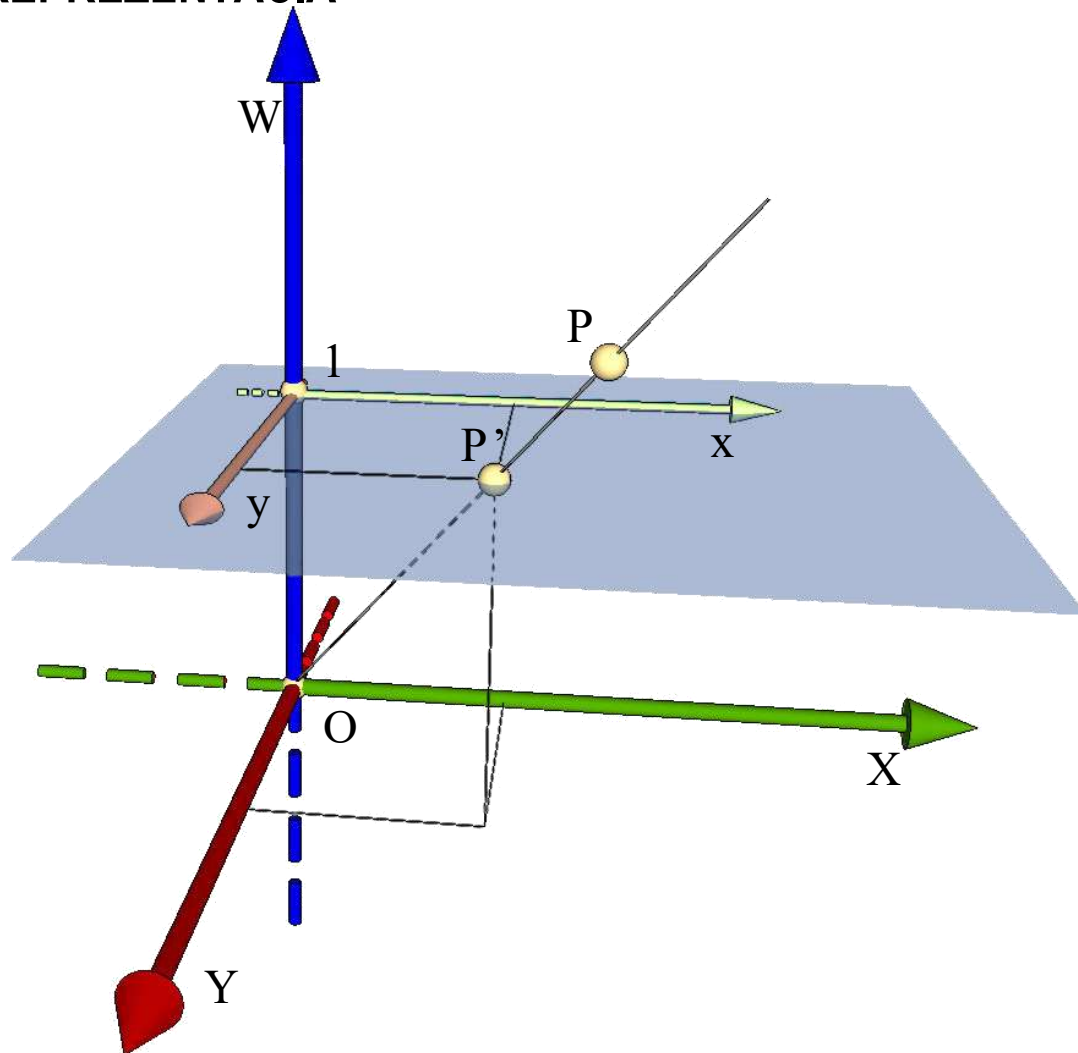
HOMOGÉNNE SÚRADNICE

$$\varphi(X, Y, W) = \begin{cases} \begin{pmatrix} \frac{X}{W} & \frac{Y}{W} \end{pmatrix} & \text{ak } W \neq 0 \\ \text{smer. } (X, Y) & \text{ak } W = 0 \end{cases}$$

$$\begin{aligned} [xw, yw] &\equiv [x, y, w] \\ [xw, yw, zw] &\equiv [x, y, z, w] \end{aligned} \quad \text{alebo} \quad \begin{aligned} \begin{bmatrix} \frac{x}{w} & \frac{y}{w} \end{bmatrix} &\equiv [x, y, w] \\ \begin{bmatrix} \frac{x}{w} & \frac{y}{w} & \frac{z}{w} \end{bmatrix} &\equiv [x, y, z, w] \end{aligned} \quad \begin{array}{l} \text{pre 2D} \\ \text{pre 3D} \end{array}$$

HOMOGÉNNE SÚRADNICE

GEOMETRICKÁ REPREZENTÁCIA



AFINNÉ TRANSFORMÁCIE

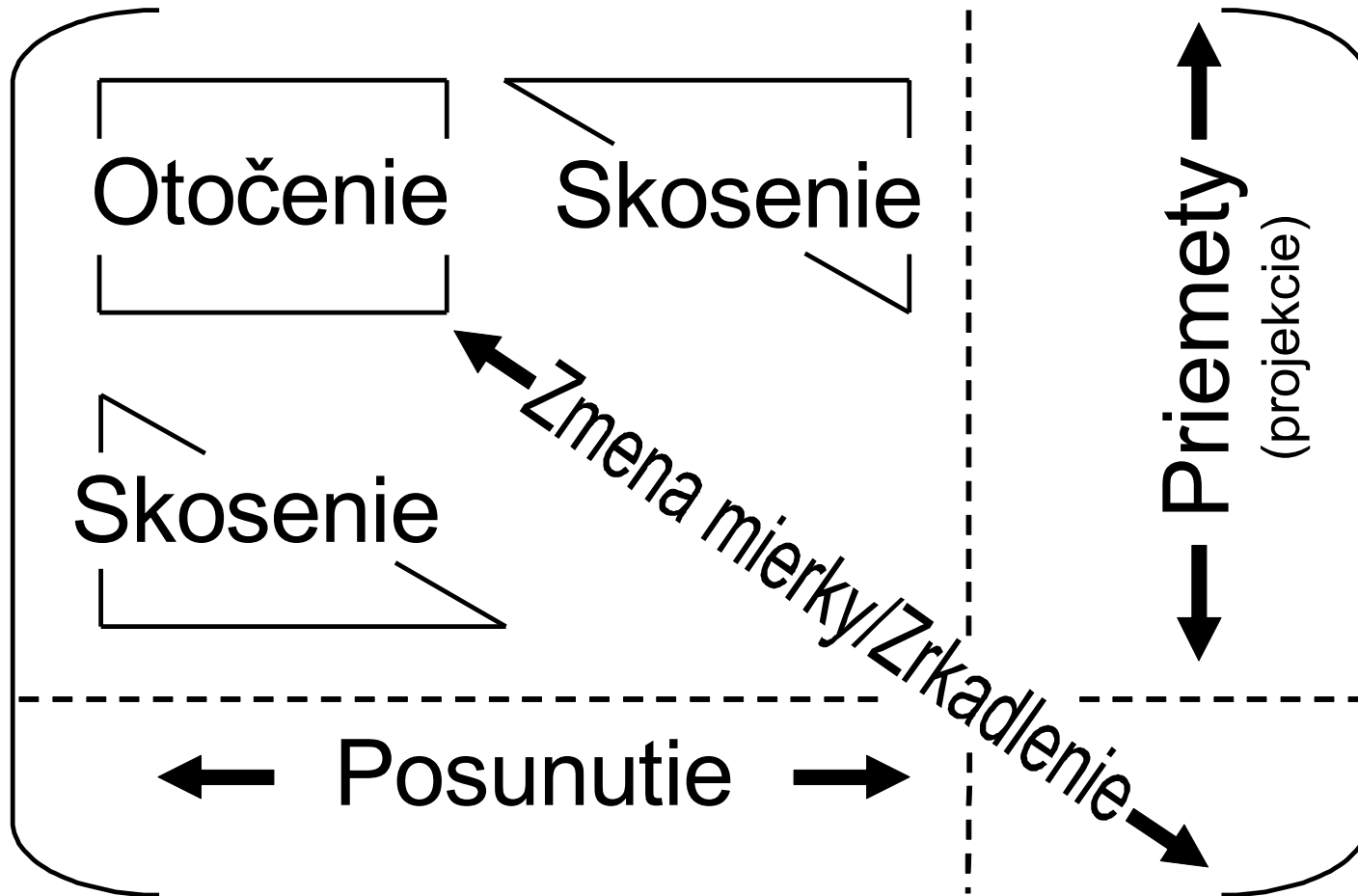
- Zrkadlenie
- Zmena mierky
- Posunutie
- Skosenie
- Otočenie



Afinná transformácia je typ lineárnej geometrickej transformácie, ktorá zachováva typ objektu a vzťah atribútov napr. rovinnosti, rovnobežnosti a pomerov vzdialeností, ale nemusí zachovávať hodnoty atribútov napr. uhly ani dĺžky. Oproti základnej lineárnej transformácii (vyjadrenej voči počiatku súradnicovej sústavy) umožňuje implementovať aj transláciu.

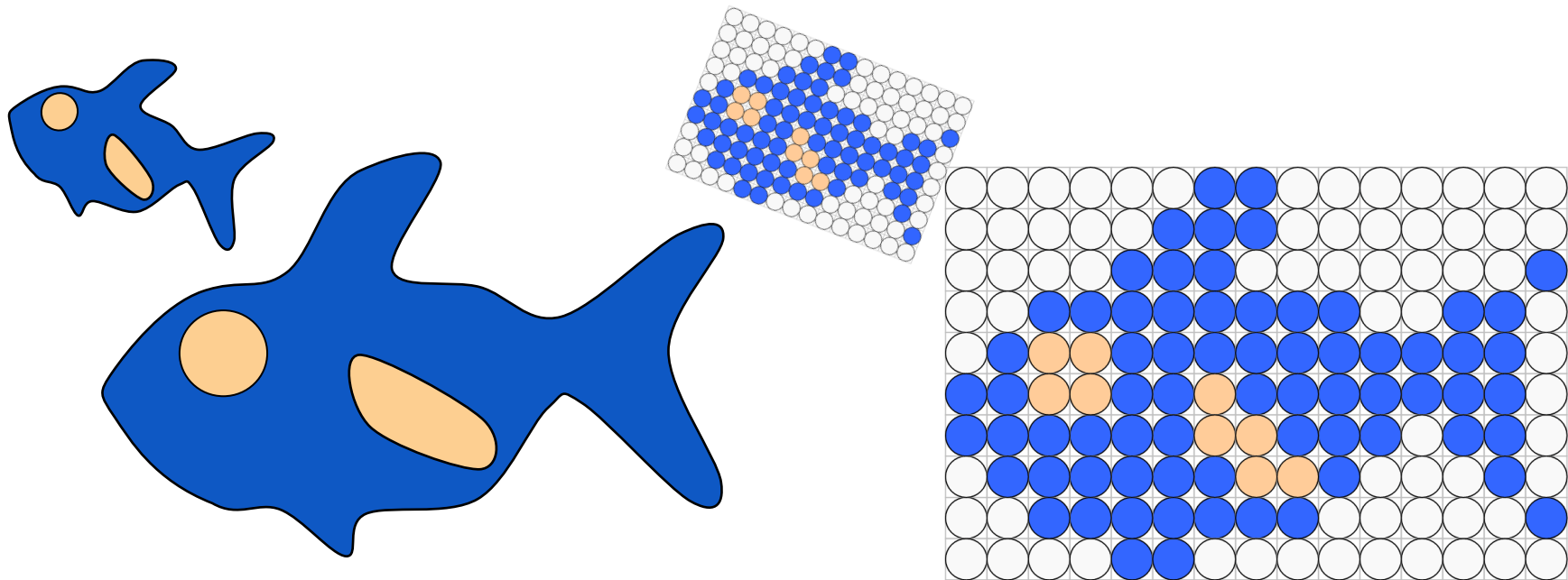


ROZMIESTNENIE PARAMETROV TRANSFORMAČNÝCH MATÍC



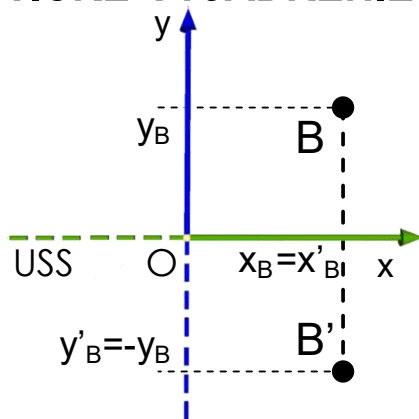
IMPLEMENTÁCIA TRANSFORMÁCIÍ

- Implementácia pre vektorové objekty
- Implementácia pre rastrové objekty



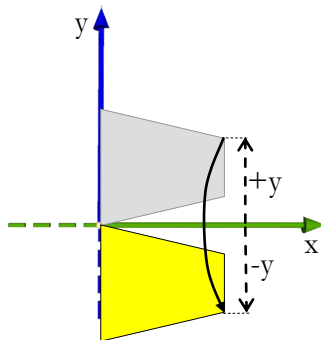
ZRKADLENIE (2D)

ANALYTICKÉ VYJADRENIE

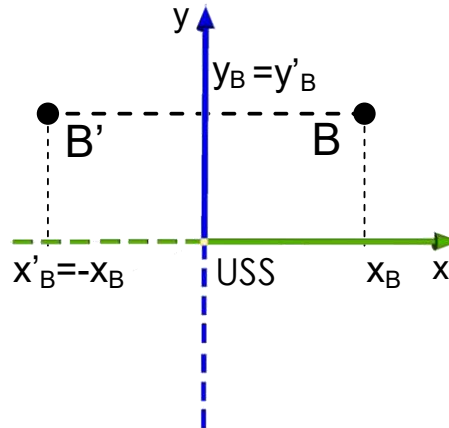


$$x'_B = x_B$$

$$y'_B = -y_B$$

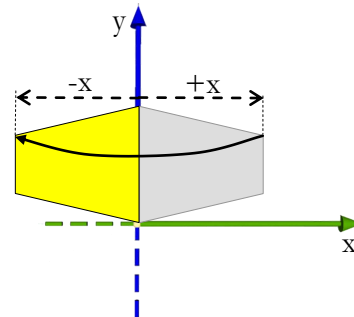


vzhľadom na os x

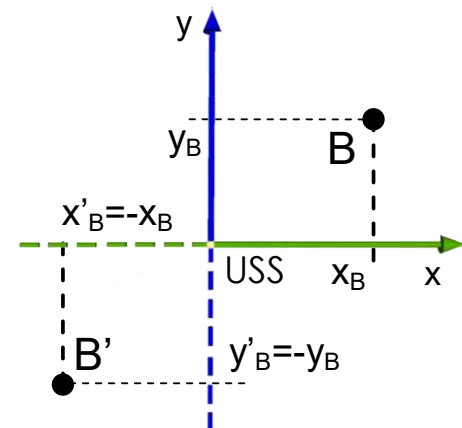


$$x'_B = -x_B$$

$$y'_B = y_B$$

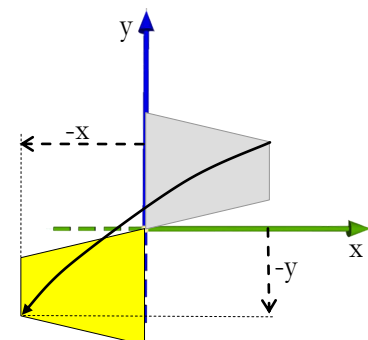


vzhľadom na os y



$$x'_B = -x_B$$

$$y'_B = -y_B$$



vzhľadom na stred

ZRKADLENIE (3D)

MATICOVÉ VYJADRENIE (HOMOGÉNNE MATICE)

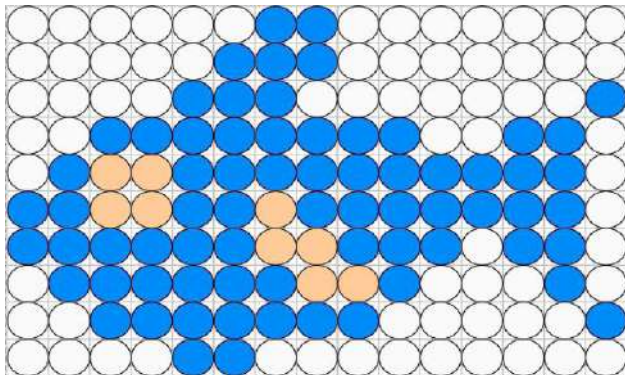
$$\mathbf{T}_{Zx} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{T}_{Zxy} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{T}_{Zs} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

vzhľadom na os x

vzhľadom na rovinu xy

vzhľadom na stred

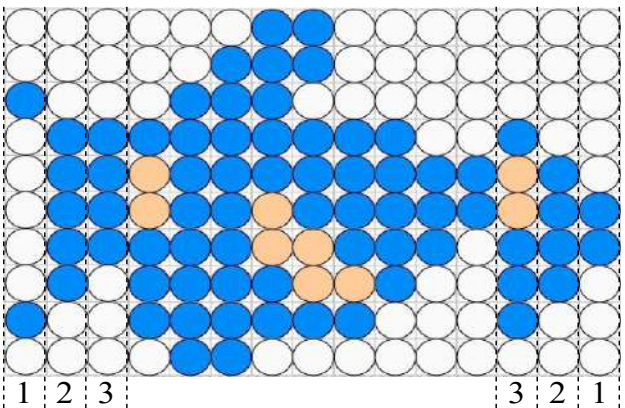
ZRKADLENIE RASTROVÝCH OBJEKTOV



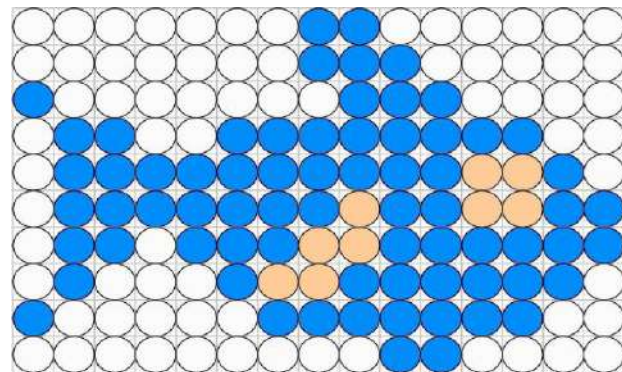
originál

0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
0	0	0	0	1	1	1	0	0	0	0	0	0	0	1
0	0	1	1	1	1	1	1	1	0	0	0	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	0	1	1	0
0	1	1	1	1	1	1	1	1	1	0	0	0	1	0
0	0	1	1	1	1	1	1	1	0	0	0	0	0	1
0	0	0	0	1	1	0	0	0	0	0	0	0	0	0

maska

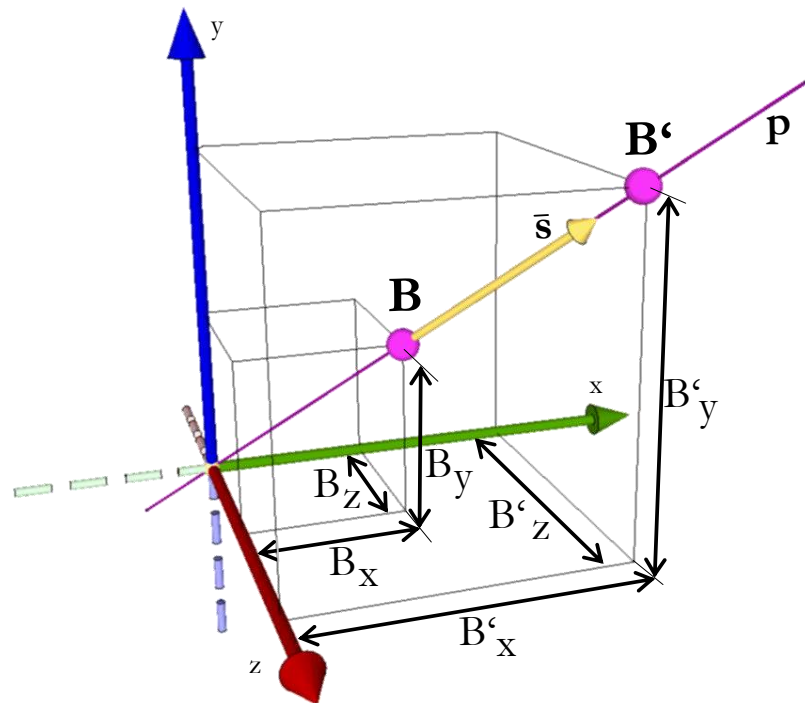
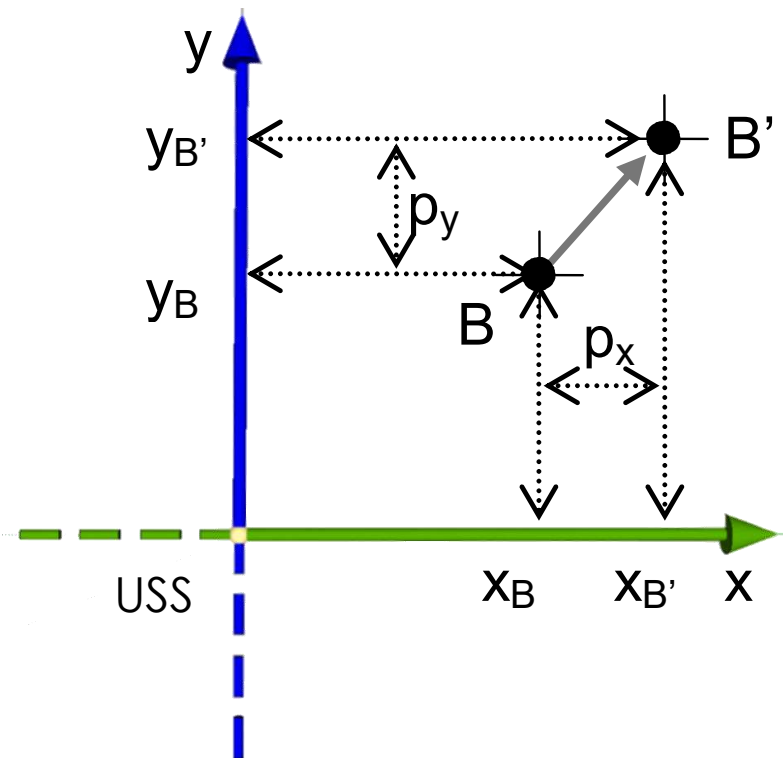


prvé 3 kroky zrkadlenia



výsledok

POSUNUTIE (TRANSLÁCIA)

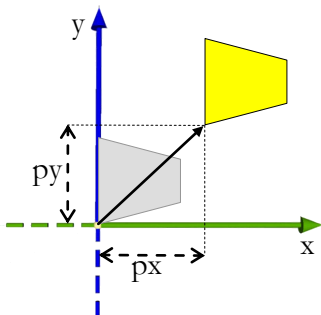


POSUNUTIE (TRANSLÁCIA)

$$x'_B = x_B + px$$

$$y'_B = y_B + py$$

$$\mathbf{T}_P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ px & py & 1 \end{bmatrix}$$

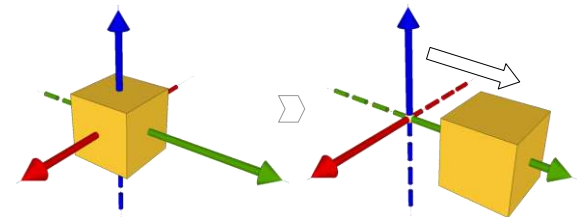


$$x'_B = x_B + px$$

$$y'_B = y_B + py$$

$$z'_B = z_B + pz$$

$$\mathbf{T}_P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ px & py & pz & 1 \end{bmatrix}$$



ZMENA MIERKY (ŠKÁLOVANIE)

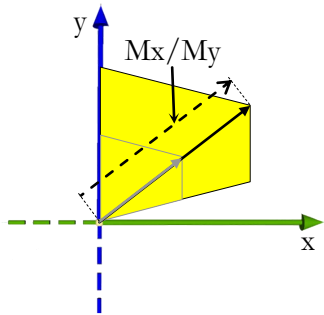
$$x'_B = M_x \times x_B$$

$$y'_B = M_y \times y_B$$

$$x'_B = M_x \times x_B$$

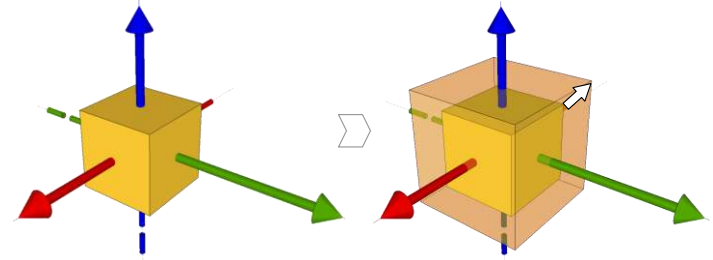
$$y'_B = M_y \times y_B$$

$$z'_B = M_z \times z_B$$



$$\mathbf{T}_M = \begin{bmatrix} M_x & 0 & 0 \\ 0 & M_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_M = \begin{bmatrix} M_x & 0 & 0 & 0 \\ 0 & M_y & 0 & 0 \\ 0 & 0 & M_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Ak sa pri zmene mierke aplikuje rovnaká hodnota koeficientu pre všetky osi, potom je takáto transformácia uniformná

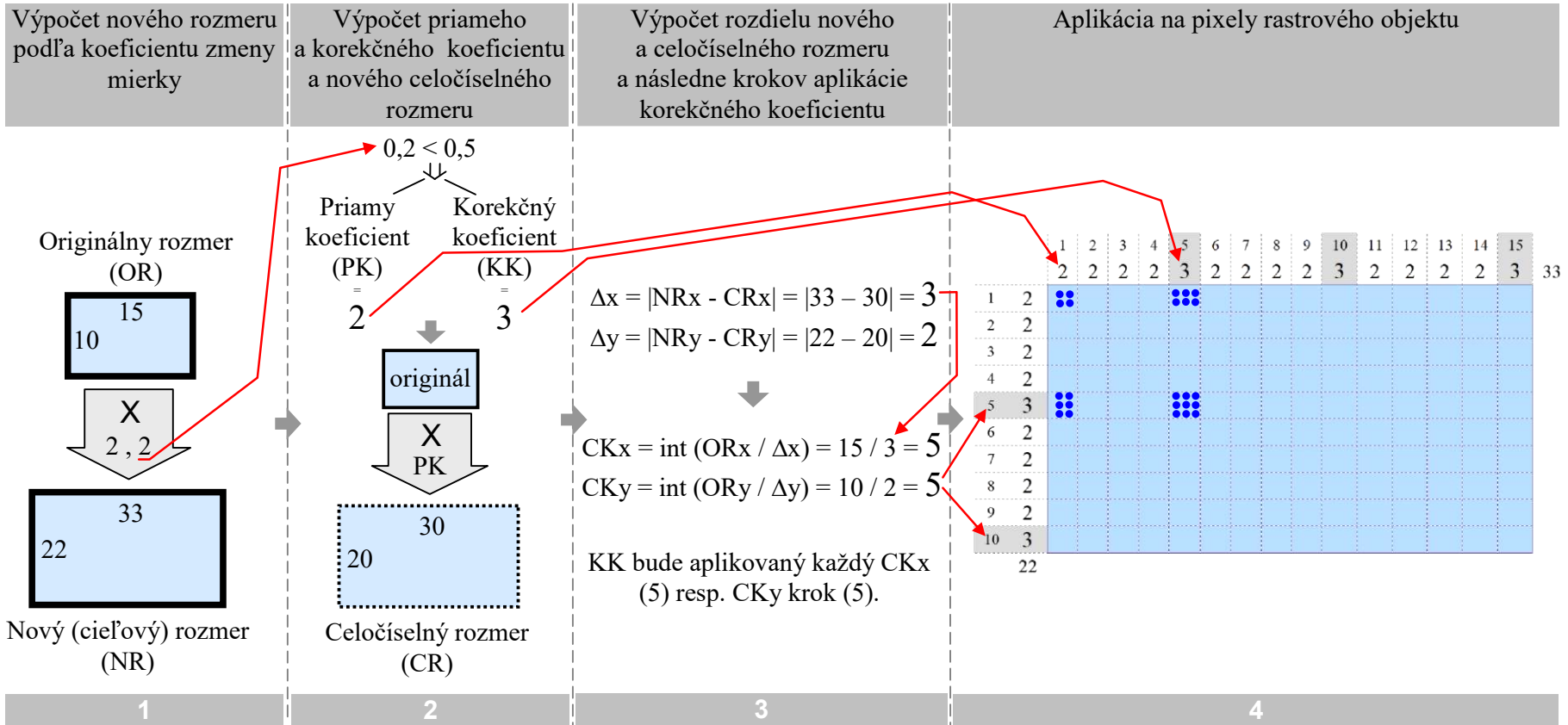
ZMENA MIERKY RASTROVÉHO OBJEKTU

ZVÄČŠENIE

1. vypočíta sa predpokladaný nový rozmer (NR) rastra
2. určí sa celočíselná a desatinná časť koeficientu zmeny mierky
3. vykoná sa test, či desatinná časť koeficientu zmeny mierky je väčšia ako 0.5.
4. určí sa pomocný výpočtový nový priamy koeficient (PK). Tento nový koeficient je buď totožný s celočíselnou časťou koeficientu zmenu mierky (ak desatinná časť koeficientu zmeny mierky $M \leq 0.5$) alebo je o 1 väčší (ak desatinná časť koeficientu zmeny mierky $M > 0.5$). Potom nový korekčný koeficient (KK) je určený presne opačne ako priamy koeficient.
5. následne sa vypočíta veľkosť rozmeru rastra (CR), ak by bol koeficient zmeny mierky len celá časť M.
6. zistí sa celočíselný rozdiel (Δ) medzi výpočtovým novým rozmerom a celočíselným rozmerom aby sa zistilo, koľko bodov je nutné ešte doplniť.
7. vypočíta sa korekčný krok (CK) (z originálneho počtu krokov t.j. originálneho rozmeru), ktorý bude určovať, kedy sa vykoná zmena celočíselného koeficientu zmeny mierky o 1 (+ alebo – podľa toho či nebola alebo bola desatinná časť koeficientu zmeny mierky > 0.5), teda kedy sa vlastne použije korekčný koeficient KK.
8. nakoniec sa vykoná konečné priradenie celočíselnej štandardnej (priamej) hodnoty koeficienta zmeny mierky a celočíselného korekčného koeficienta v jednotlivých krokoch.

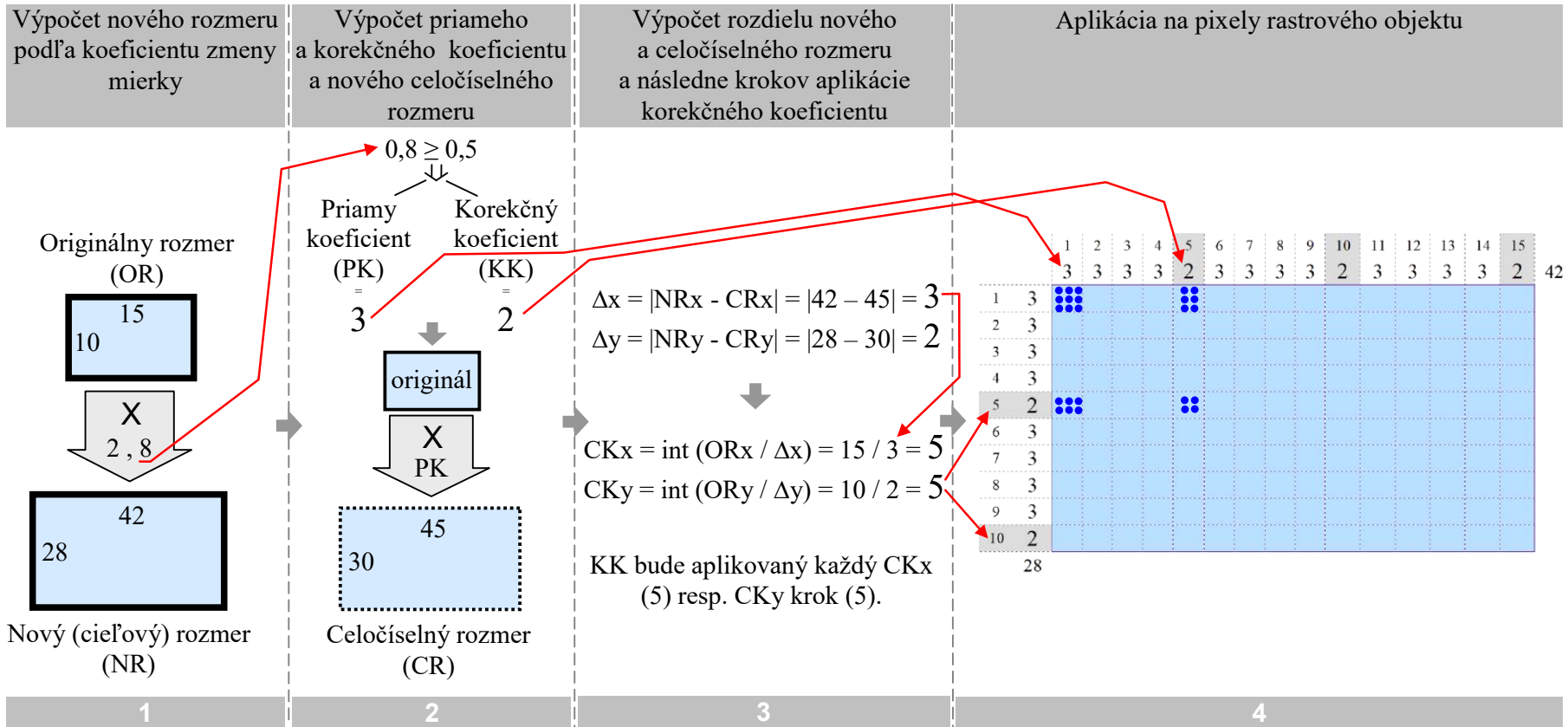
ZMENA MIERKY RASTROVÉHO OBJEKTU

ZVÄČŠENIE, DESATINNÁ ČASŤ KOEFICIENTU <0.5



ZMENA MIERKY RASTROVÉHO OBJEKTU

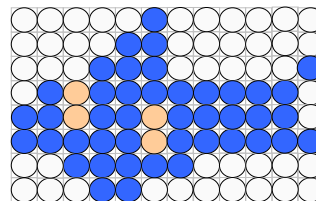
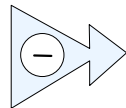
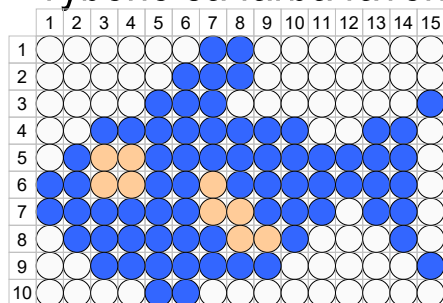
ZVÄČŠENIE, DESATINNÁ ČASŤ KOEFICIENTU ≥ 0.5



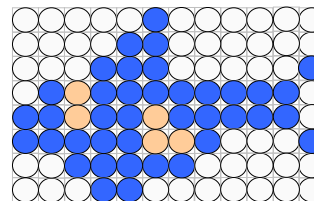
ZMENA MIERKY RASTROVÉHO OBJEKTU

ZMENŠENIE, URČENIE FARBY

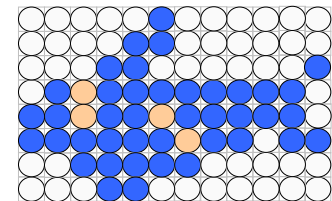
1. Vypočíta sa zmena mierky v opačnom smere t.j. ako pri zväčšení len s $M = 1/M$, tým sa zistí, vlastne aká matica pixelov bude tvoriť subpixel (napr. 2x2, 3x3 či 3x2).
2. Následne sa určí farba subpixelu:
 - spriemernením farieb pixelov alebo použitím mediánovej funkcie v matici pixelov.
 - zistí sa početnosť výskytu farieb v matici pixelov a vyberie sa tá farba, ktorá sa vyskytuje najčastejšie. Ak je výskyt farieb rovnaký, vyberie sa farba podľa iného pravidla alebo ľubovoľná z vyskytujúcich sa farieb.
 - vyberie sa farba, ktorá sa vyskytuje najmenej krát.
 - vyberie sa farba ľavého horného bodu obdĺžnika



ľavý horný roh

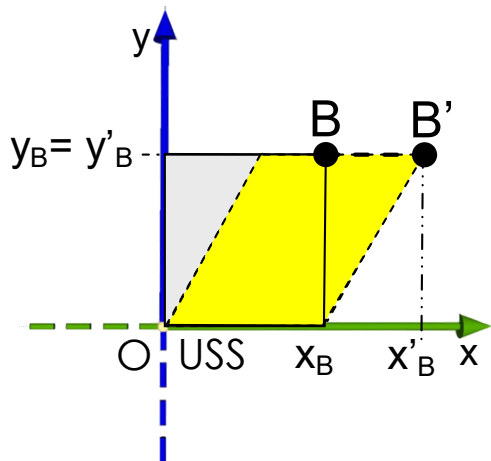


väčšinový výskyt, ak rovnako, vyberie sa svetlejšia

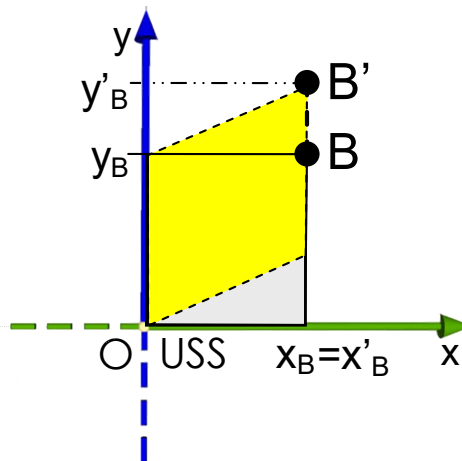


menšinový výskyt, ak rovnako, vyberie sa svetlejšia

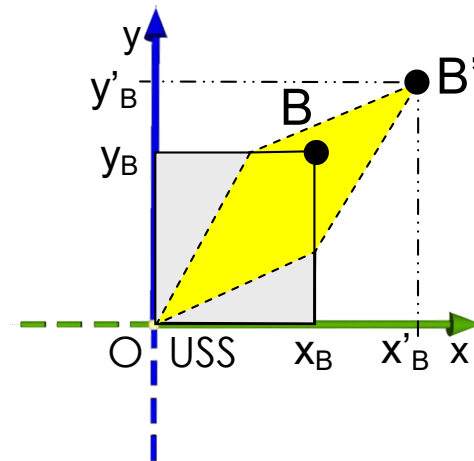
SKOSENIE (2D, 3D)



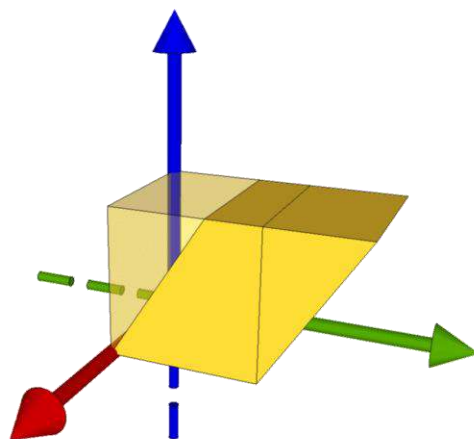
skosenie v smere osi x



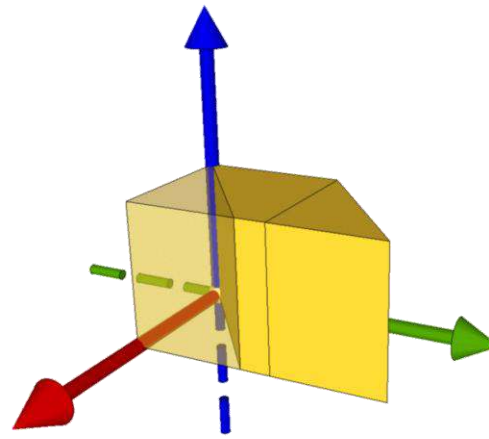
skosenie v smere osi y



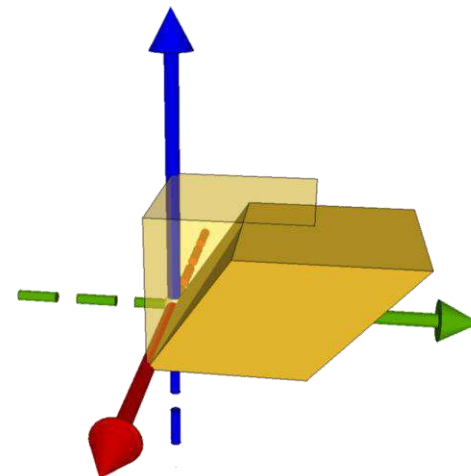
skosenie v oboch smeroch



skosenie v smere osi x
vzhľadom na z



skosenie v smere osi x
vzhľadom na y



skosenie v smere
roviny xy

SKOSENIE (2D)

$$x'_B = x_B + S_x \times y_B$$

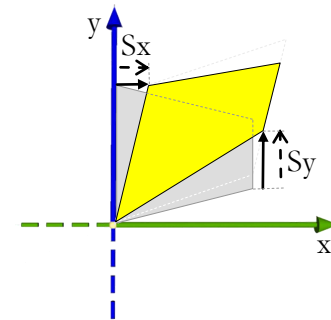
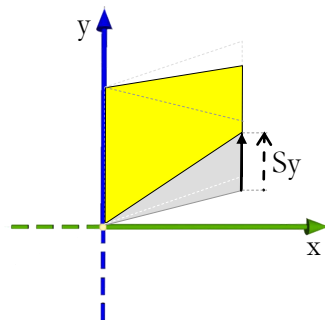
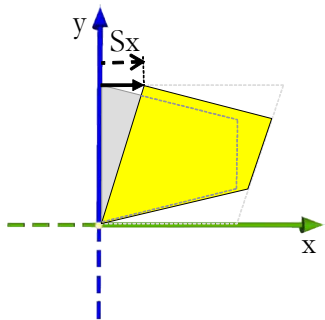
$$x'_B = x_B$$

$$y'_B = y_B$$

$$y'_B = y_B + S_y \times x_B$$

$$\mathbf{T}_{S_x} = \begin{bmatrix} 1 & 0 & 0 \\ S_x & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_{S_y} = \begin{bmatrix} 1 & S_y & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

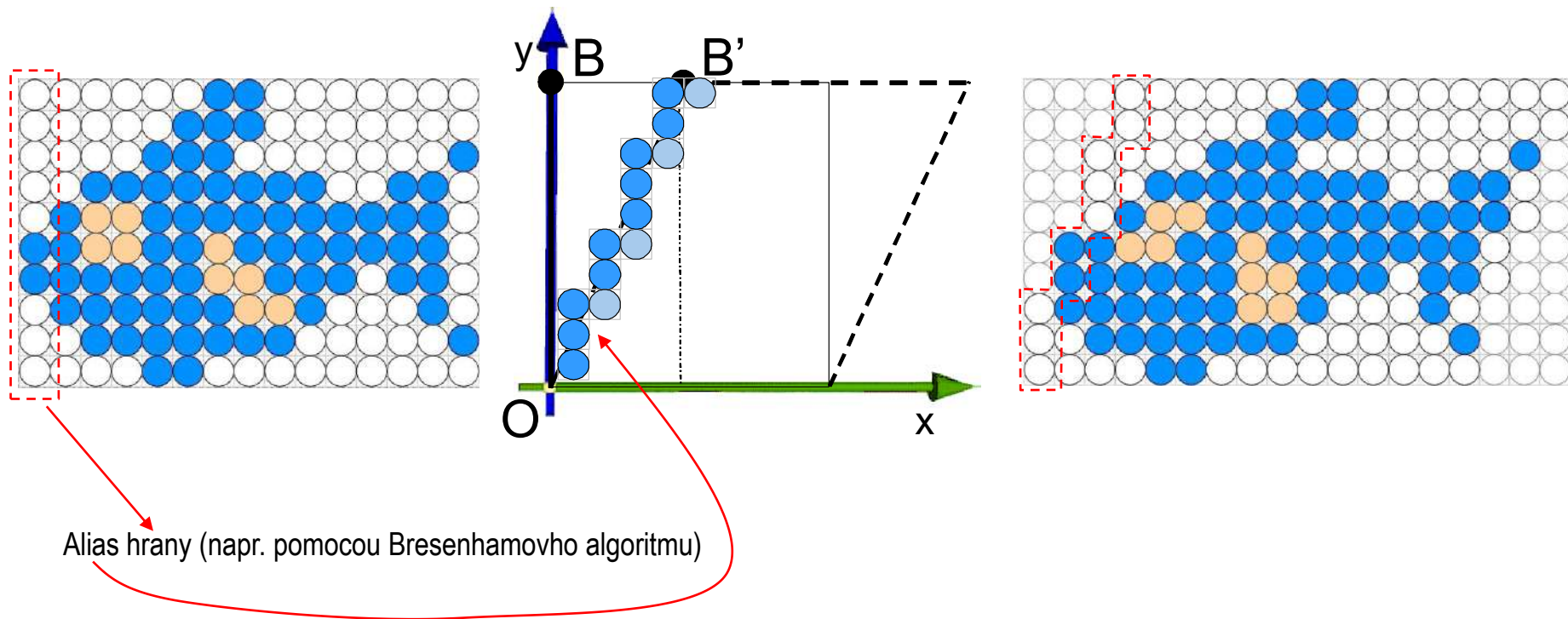


SKOSENIE (3D)

$$\begin{array}{lll}
 x'_B = x_B + S_{xoz} \times z_B & x'_B = x_B + S_{xoy} \times y_B & x'_B = x_B + S_{xoz} \times z_B \\
 y'_B = y_B & y'_B = y_B & y'_B = y_B + S_{yoz} \times z_B \\
 z'_B = z_B & z'_B = z_B & z'_B = z_B
 \end{array}$$

$$\mathbf{T}_{Sxoz} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ S_{xoz} & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{T}_{Sxoy} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ S_{xoy} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{T}_{Sxy} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ S_{xoz} & S_{yoz} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

SKOSENIE RASTROVÝCH OBJEKTOV



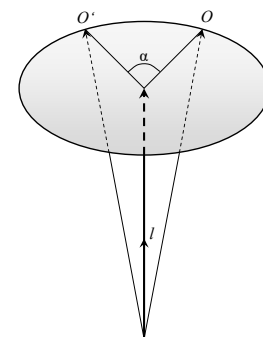
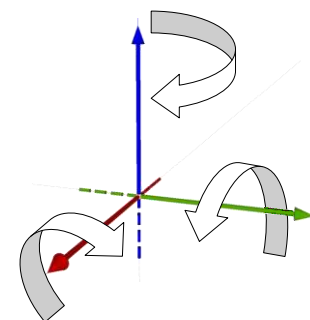
Ak je potrebné skosenie v oboch smeroch, musí sa vykonať postupne, najprv v prvom a potom v druhom smere.

OTÁČANIE (ROTÁCIA, 3D)

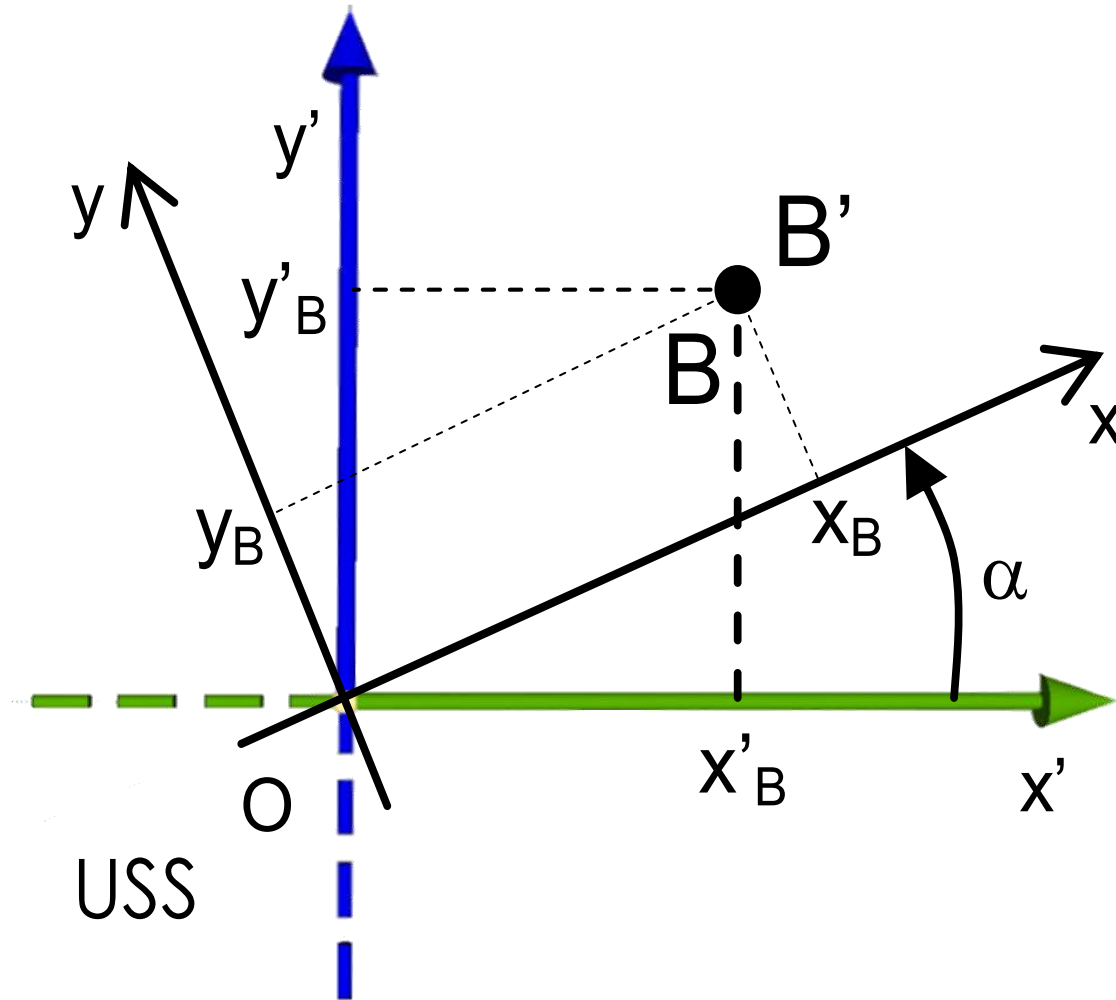
- otáčanie definované **Eulerovými uhlami** a reprezentované všeobecnými transformačnými maticami. V princípe sa jedná o rozklad všeobecného otočenia na tri zložky a to okolo jednotlivých osí karteziánskej súradnicovej sústavy.
- otáčanie definované **Eulerovým teorémom** a reprezentované *quaterniónmi*.



Leonhard Euler



OTÁČANIE (ROTÁCIA 2D)

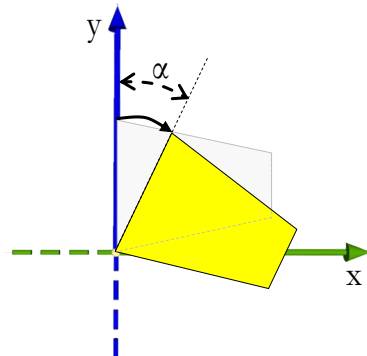


OTÁČANIE (ROTÁCIA, 2D)

$$x'_B = x_B \times \cos(\alpha) - y_B \times \sin(\alpha)$$

$$y'_B = x_B \times \sin(\alpha) + y_B \times \cos(\alpha)$$

$$\mathbf{T}_O = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



OTÁČANIE (ROTÁCIA, 3D)

$$x'_B = x_B$$

$$y'_B = y_B \times \cos(\alpha) - z_B \times \sin(\alpha)$$

$$z'_B = y_B \times \sin(\alpha) + z_B \times \cos(\alpha)$$

$$x'_B = x_B \times \cos(\alpha) + z_B \times \sin(\alpha)$$

$$y'_B = y_B$$

$$z'_B = -x_B \times \sin(\alpha) + z_B \times \cos(\alpha)$$

$$x'_B = x_B \times \cos(\alpha) - y_B \times \sin(\alpha)$$

$$y'_B = x_B \times \sin(\alpha) + y_B \times \cos(\alpha)$$

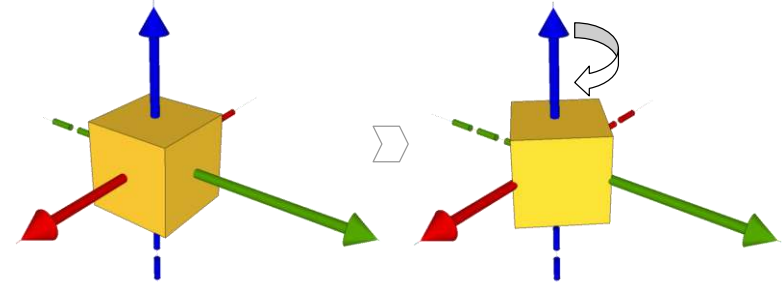
$$z'_B = z_B$$

OTÁČANIE (ROTÁCIA, 3D)

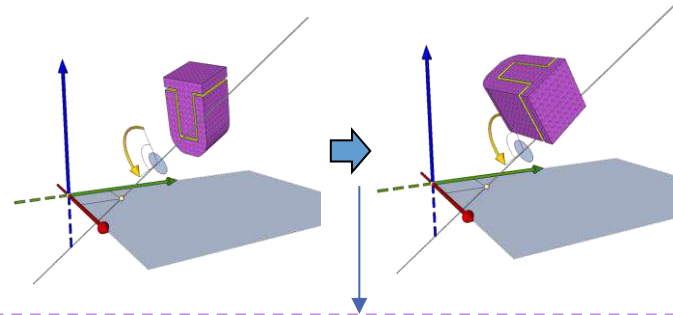
$$\mathbf{T}_{Ox} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_{Oz} = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 & 0 \\ -\sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

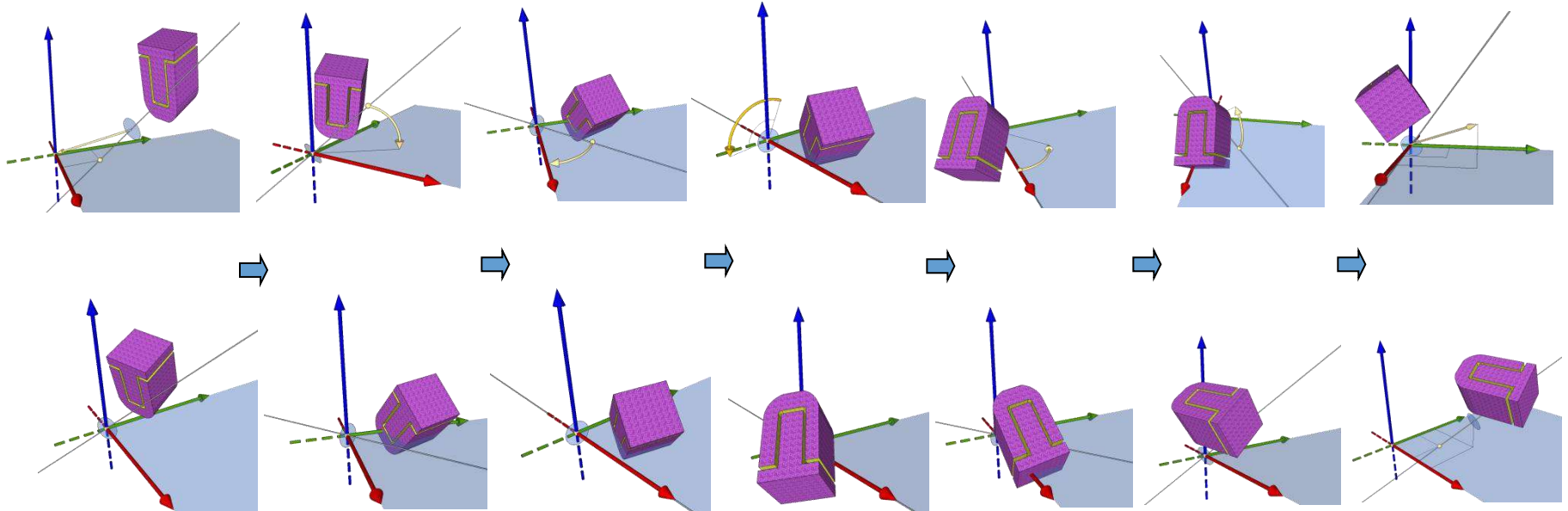
$$\mathbf{T}_{Oy} = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



OTÁČANIE OKOLO VŠEOBECNEJ PRIAMKY

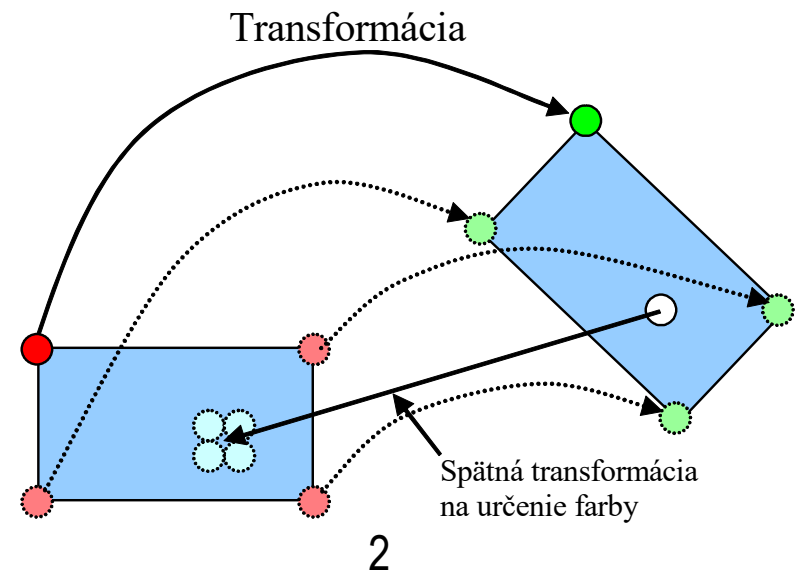
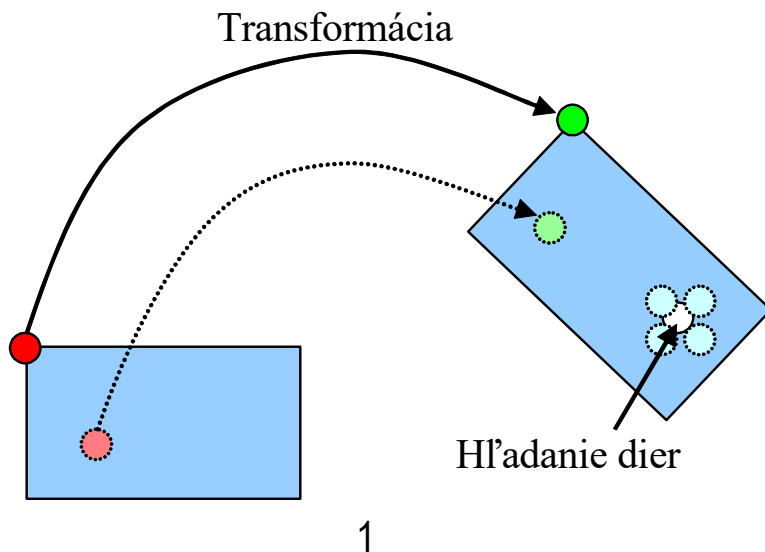


$$T_P \times T_{Oa} \times T_{Ob} \times T_O \times T_{Ob}^{-1} \times T_{Oa}^{-1} \times T_P^{-1}$$



OTÁČANIE RASTROVÝCH OBJEKTOV

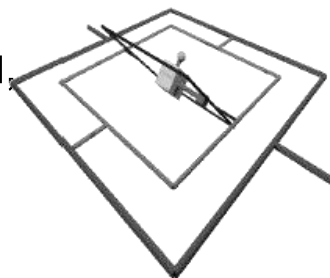
1. *Priame otáčanie* - Otočenie s interpoláciou medziľahlých bodov
2. *Spätné otáčanie* - Otočenie s interpoláciou všetkých bodov



OTÁČANIE A STRATA STUPŇA VOĽNOSTI

Je ťažké predvídať ako sa postupné rotácie okolo základných osí navzájom ovplyvnia. (maticová reprezentácia Eulerových uhlov má prirodzenú jedinečnosť v parametrizácii) Je možné vytvoriť takú postupnosť rotácií, že vo výslednej rotácii sa stratí jeden stupeň voľnosti. Táto situácia sa nazýva **gimbal lock** (*strata stupňa voľnosti*).

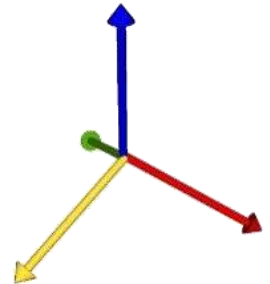
Gimbal lock je pojem prevzatý z leteckého a vesmírneho priemyslu, kde sa používajú gyroskopy (vytvárajú umelý horizont)



ROTÁCIA POMOCOU QUATERNIÓNOV

Pre popis rotácie (otočenia) nasledovaného prípadnou zmenou mierky sú potrebné štyri čísla (W. R. Hamilton, 1843, aplikácia v PG, K. Shoemake, 1985) :

- jedno číslo popisuje veľkosť zmeny mierky,
- jedno veľkosť uhla (v stupňoch), o ktorý sa bude otáčať, a
- posledné dve čísla označujú rovinu, v ktorej sa bude vektor otáčať
- Quaternión je definovaný ako komplexné číslo
 - $\mathbf{q} = w + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$
- Rotácia je nahradená násobením quaterniónov, kde
 - $\mathbf{i} * \mathbf{j} = -\mathbf{j} * \mathbf{i} = \mathbf{k}$
 - $\mathbf{j} * \mathbf{k} = -\mathbf{k} * \mathbf{j} = \mathbf{i}$
 - $\mathbf{k} * \mathbf{i} = -\mathbf{i} * \mathbf{k} = \mathbf{j}$



APLIKÁCIE TRANSFORMÁCIÍ

MORPHING, WARPING (2D)

- **Warping** – je *statická deformácia* (transformácia) objektu, aby sa zmenil jeho tvar, nemení však identitu objektu. Hodnoty parametrov je možné získať z iného objektu. V PG sa používa napríklad na korekciu perspektívy či mapovanie textúr na 3D objekty a pod.
- **Morphing** - je metóda používaná na *plynulú (dynamickú) premenu* jedného objektu na druhý. Mení identitu vstupného objektu na výstupný. V PG sa používa napríklad pri animáciách alebo vizuálnych efektoch.

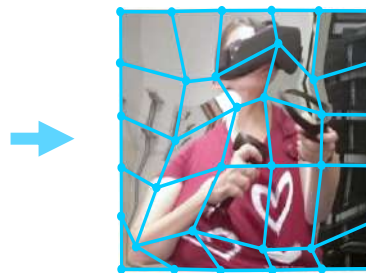


APLIKÁCIE TRANSFORMÁCIÍ

DEFORMÁCIA (WARPING, 2D)



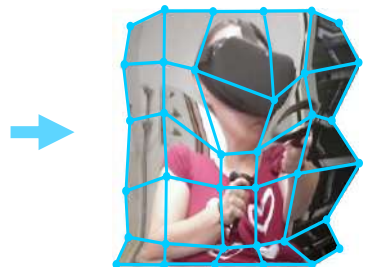
originál



Definovanie deformačnej siete



Príklady deformovania podľa parametrov siete
(identita sa nemení)



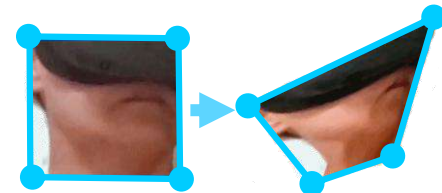
zdroj pre získanie
deformačnej siete



Získanie deformačnej siete



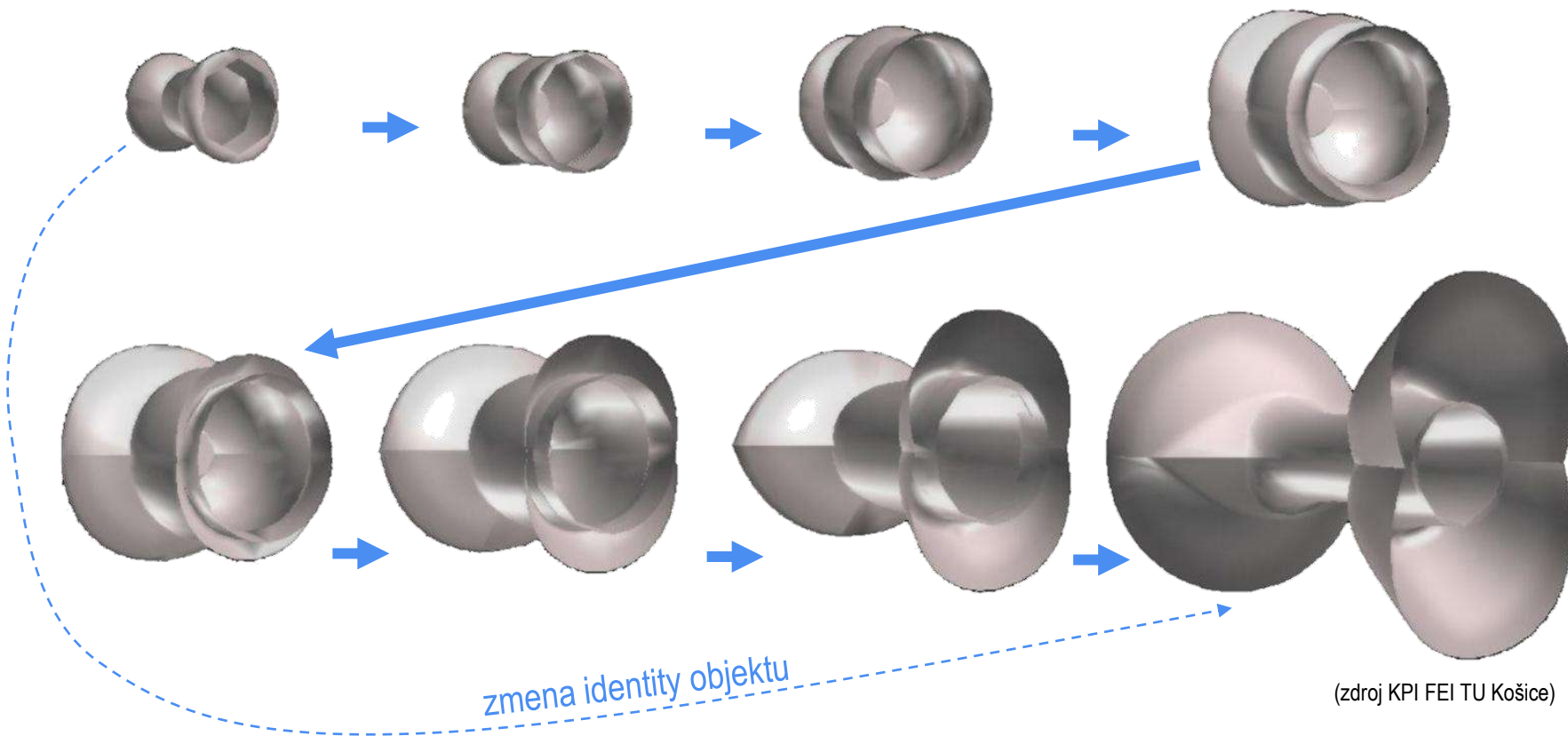
Aplikácia deformačnej siete



Detail deformovania jednej
dlaždice

APLIKÁCIE TRANSFORMÁCIÍ

MORPHING (3D)



Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

PROJEKČNÉ A ZOBRAZOVACIE TRANSFORMÁCIE

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 05

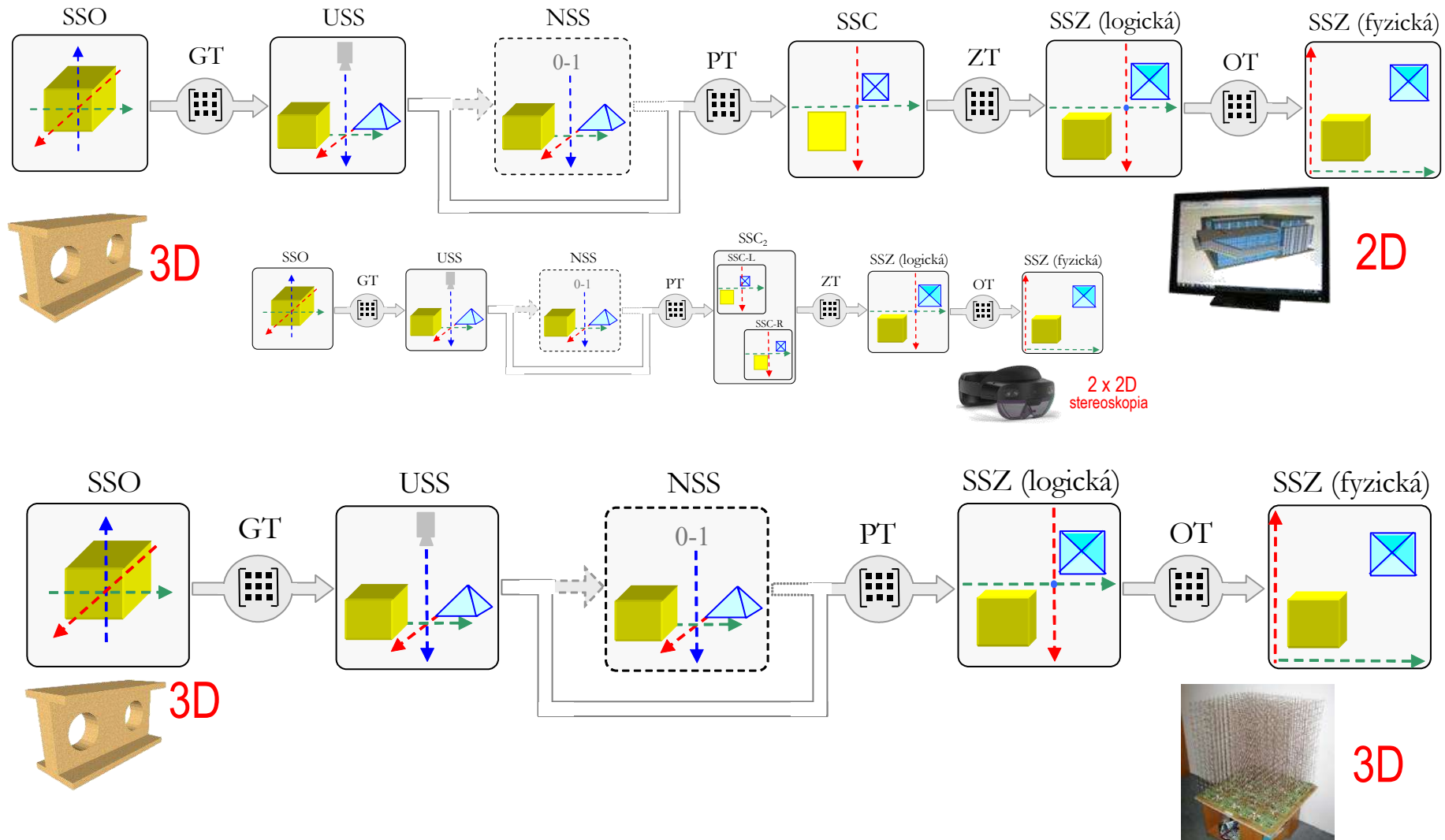
© 2025

VRSTVY VIZUALIZAČNÉHO PROCESU

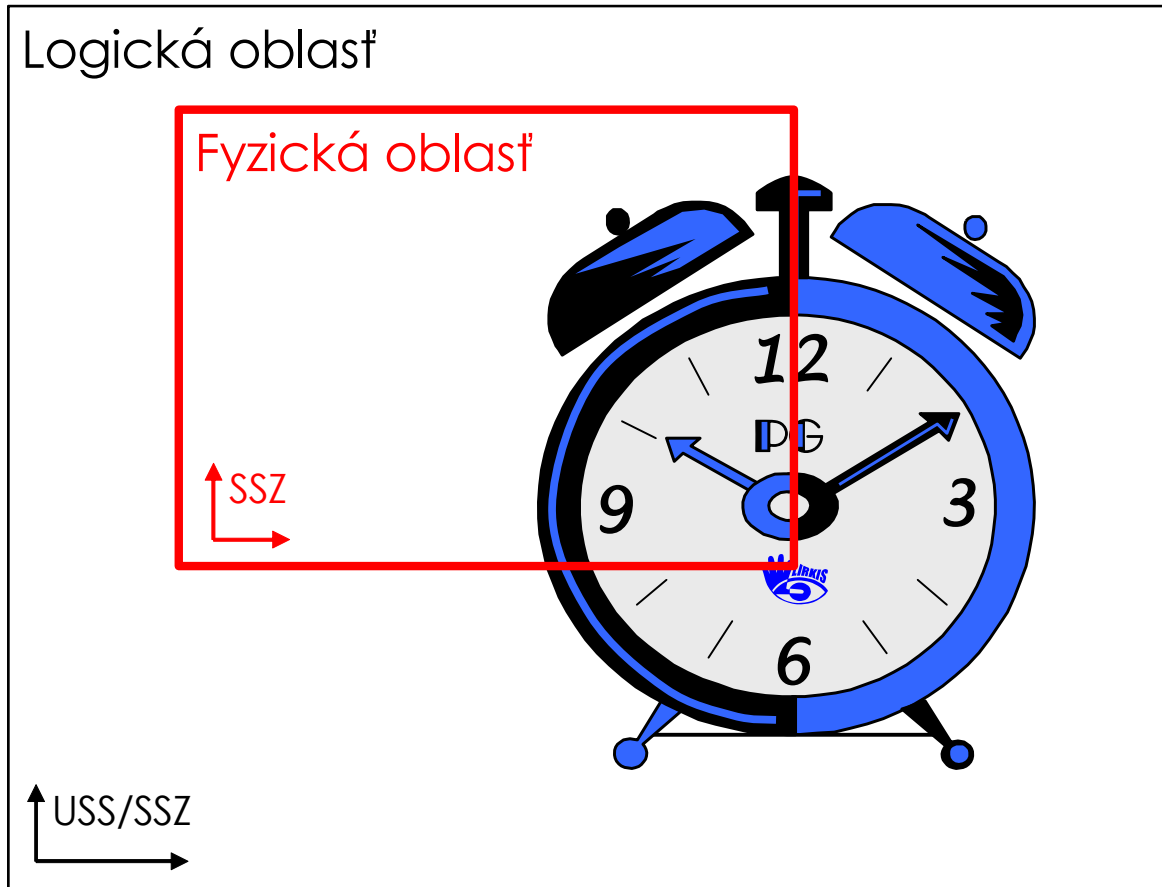
1. Definovanie/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



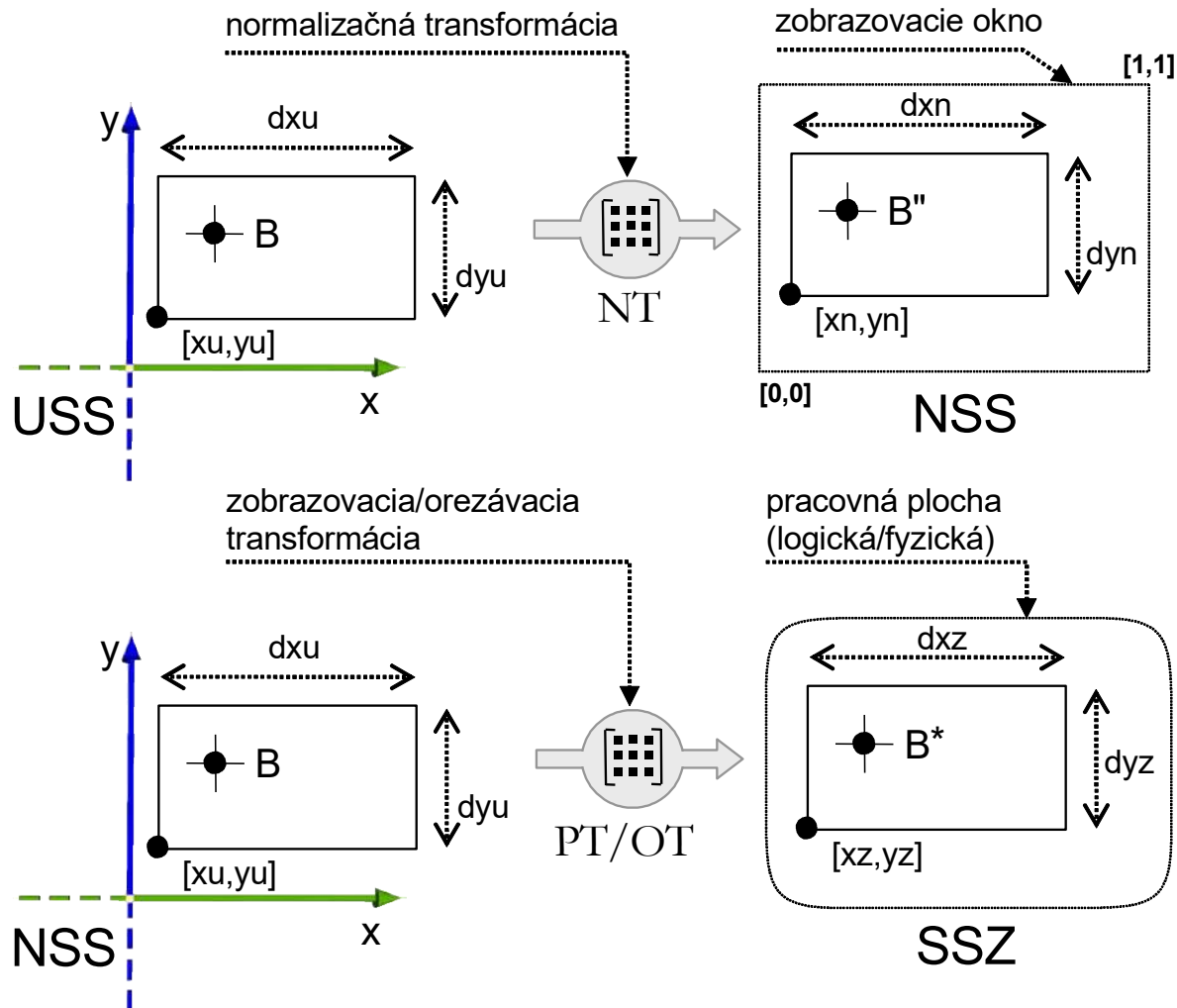
TRANSFORMAČNÉ REŤAZCE $ND \rightarrow N$ ALEBO $(N-1)D$



LOGICKÁ A FYZICKÁ PRACOVNÁ OBLASŤ



TYPY ZOBRAZOVACÍCH TRANSFORMÁCIÍ V 2D



NAJPOUŽÍVANEJŠIE OREZÁVACIE ALGORITMY

- **Cohen–Sutherlandov algoritmus**

- slúži na orezávanie úsečiek (čiar),
- funguje na princípe bitových masiek (kódov),
- používa sa väčšinou v softvérových riešeniach v pravouhlej oblasti.

- **Sutherland–Hodgmanov algoritmus**

- slúži na orezávanie konvexných polygónov,
- funguje na princípe postupného (iteratívneho) testovania každého vrcholu polygónu voči hraniciam orezávacieho okna,
- používa sa väčšinou v softvérových riešeniach v pravouhlej oblasti ale je možné sa stretnúť aj s hardvérovou implementáciou.

- **Liang–Barského algoritmus**

- podobný ako Cohen-Sutherlandov algoritmus na orezanie čiar, používa však menej výpočtov a teda je rýchlejší
- funguje na princípe použitia parametrického tvaru čiar
- používa sa väčšinou v softvérových riešeniach ale je možné sa stretnúť aj s hardvérovou implementáciou.

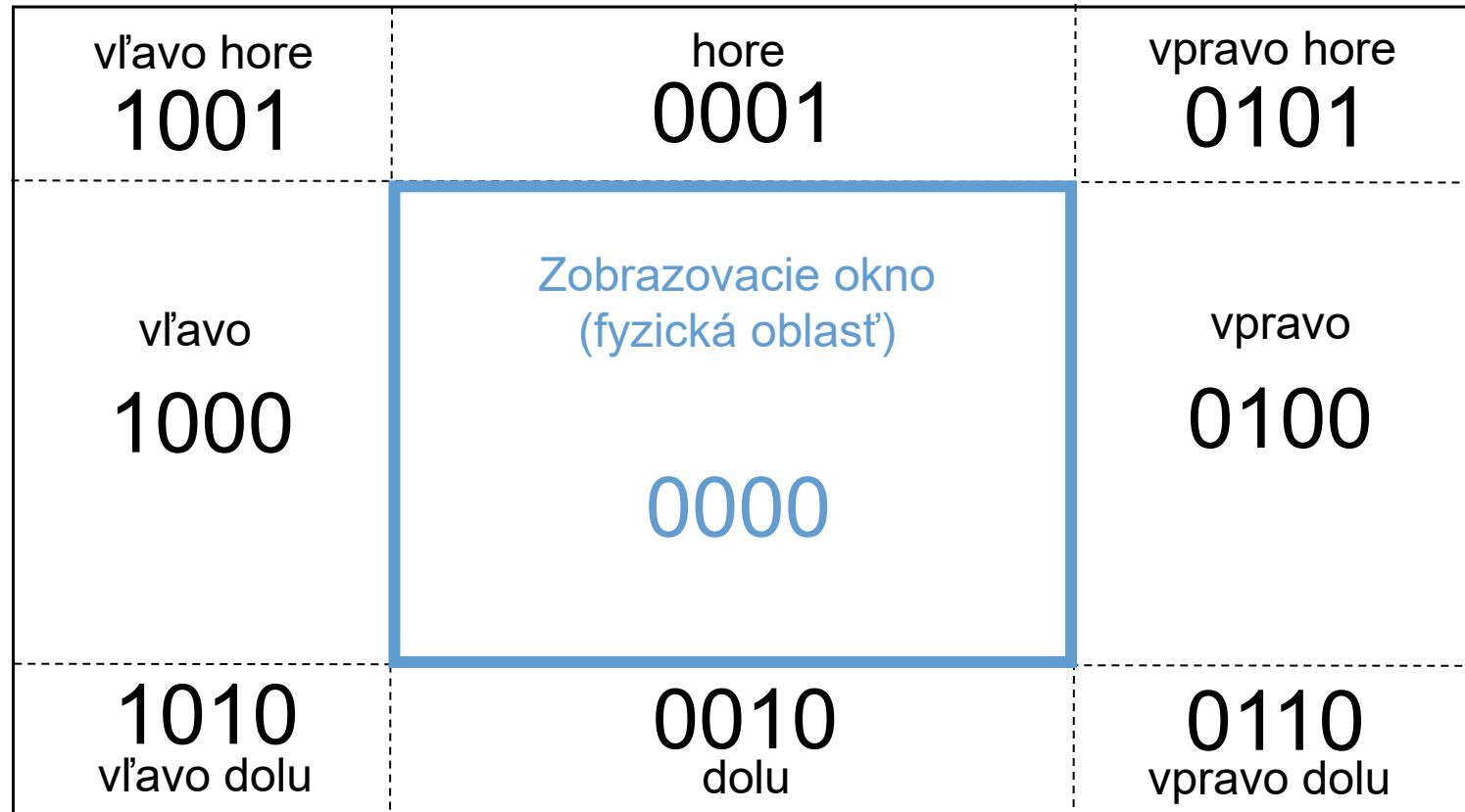
- **Weiler–Athertonov algoritmus**

- podobný ako Sutherland-Hodgmanov algoritmus, ale umožňuje aj spracovanie nekonvexných polygónov,
- používa sa menej, kvôli svojej zložitosti.



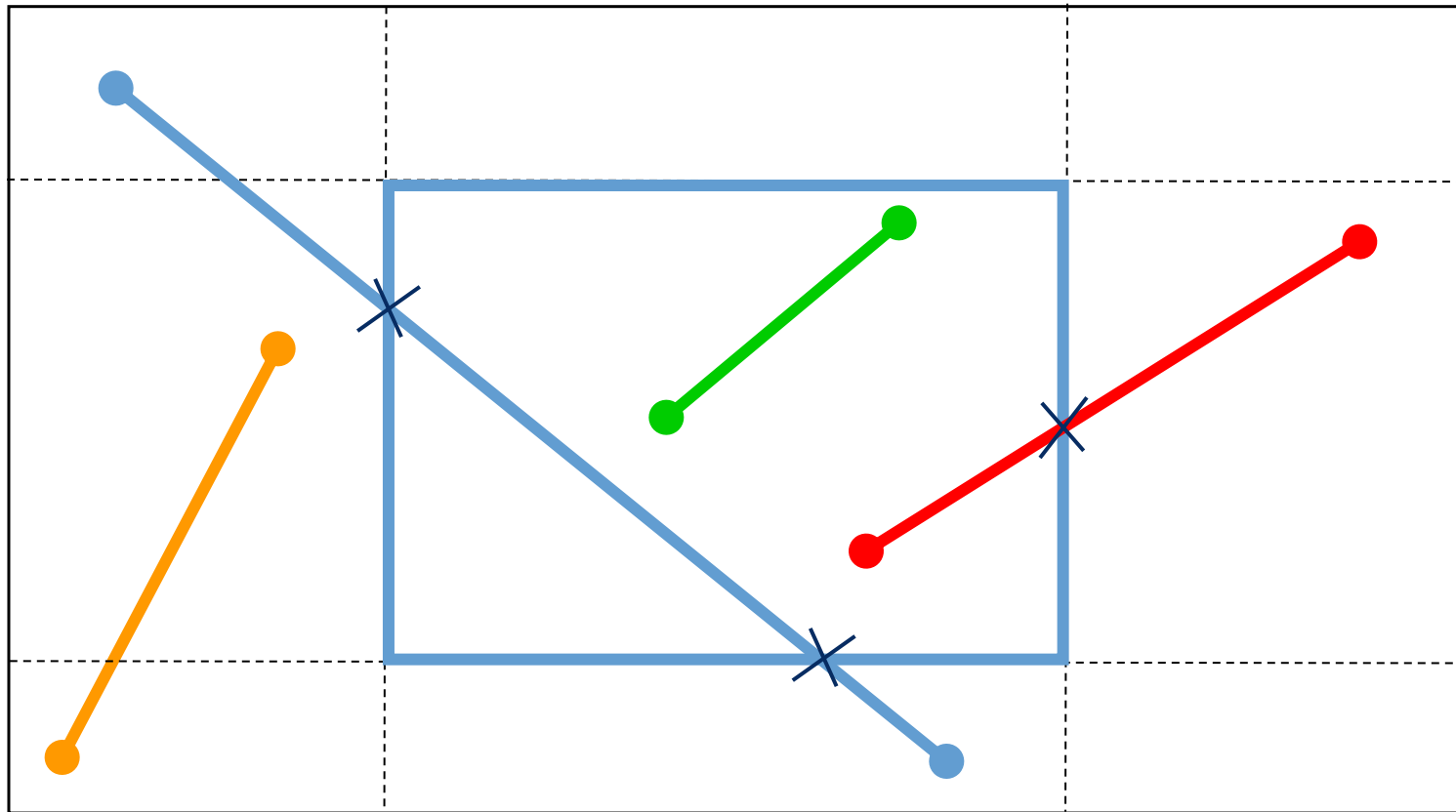
COHEN-SUTHERLANDOV ALGORITMUS

(OREZÁVACIA TRANSFORMÁCIA)



COHEN-SUTHERLANDOV ALGORITMUS

(OREZÁVACIA TRANSFORMÁCIA)

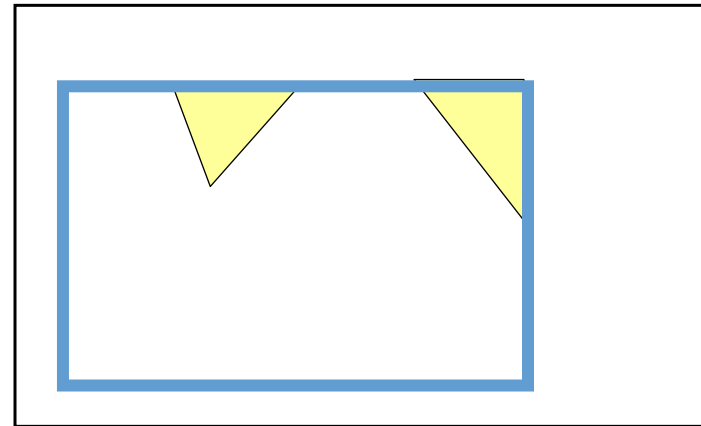
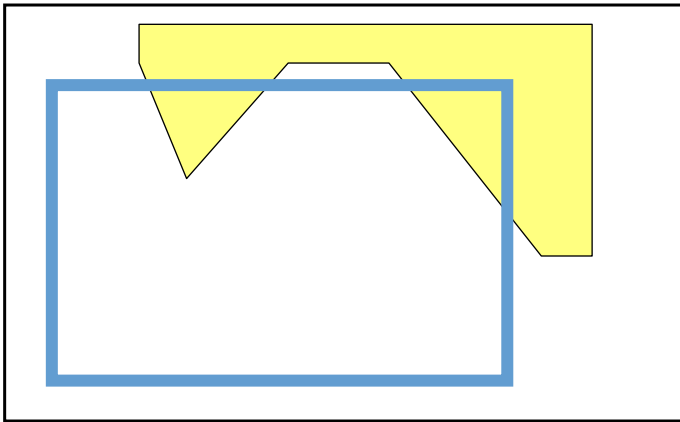


COHEN-SUTHERLANDOV ALGORITMUS

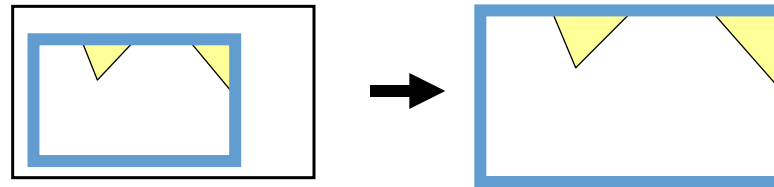
(OREZÁVACIA TRANSFORMÁCIA)

- kódy oboch koncových bodov sú nulové (prázdne) potom **oba koncové body ležia vo vnútri** zobrazovacieho okna a je možné úsečku vykresliť bez orezania.
- jeden z kódov nenulový, potom je nutné orezanie, pretože **časť úsečky určite leží mimo** zobrazovacieho okna.
- obidva kódy sú nenulové. Potom sú možné dva prípady:
 - **celá úsečka je mimo** zobrazovacieho okna a nevykreslí sa. Toto sa deje najmä ak obidva kódy dva príslušné bity rovnaké (napr. 1000 a 1010).
 - **časť úsečky je v okne**. Toto sa môže stať, ak obidva kódy nemajú dva príslušné bity rovnaké (napr. 1001 a 0010). Vtedy je nutné orezanie.

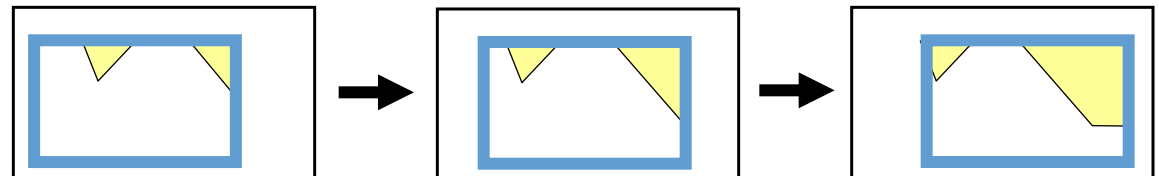
TRANSFOKÁCIA A PANORÁMOVANIE



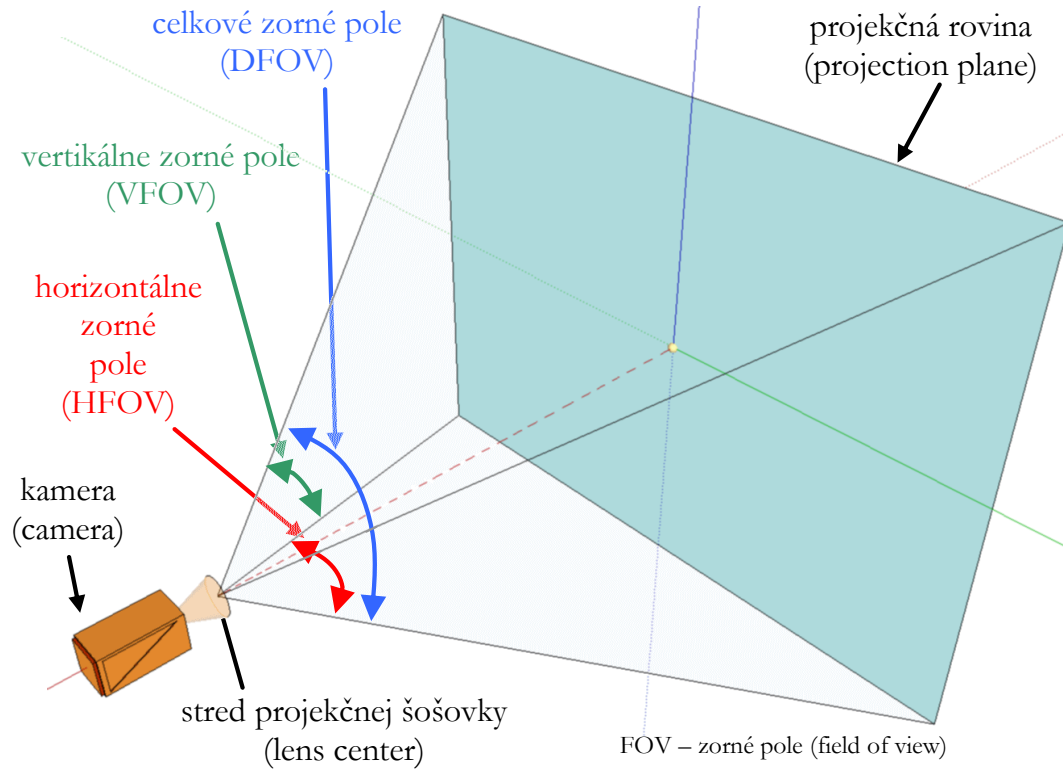
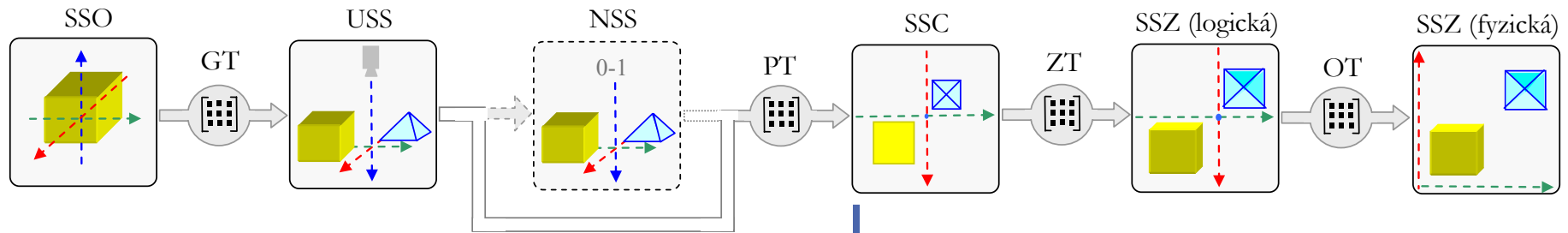
- Transfokácia



- Panorámovanie

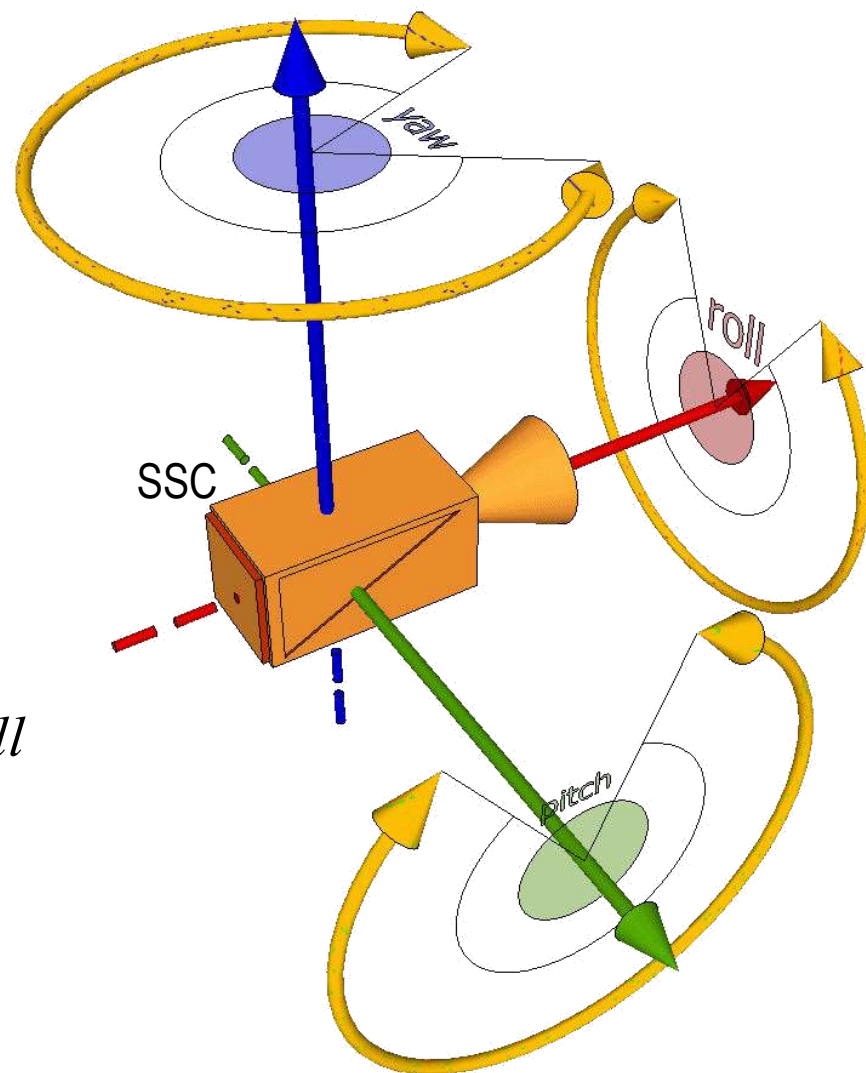


PROJEKCIA



OTÁČANIE KAMERY NA BÁZE EULEROVÝCH UHLOV ROZKLAD, ORBITA

- Otáčanie kamery v pôdoryse SSC (*yaw*)
- Sklonenie/zdvihnutie pohľadu kamery v SSC (*pitch*)
- Otáčanie okolo vektora pohľadu kamery v SSC (*roll*)



$$T_{cam} = T_{yaw} \times T_{pitch} \times T_{roll}$$

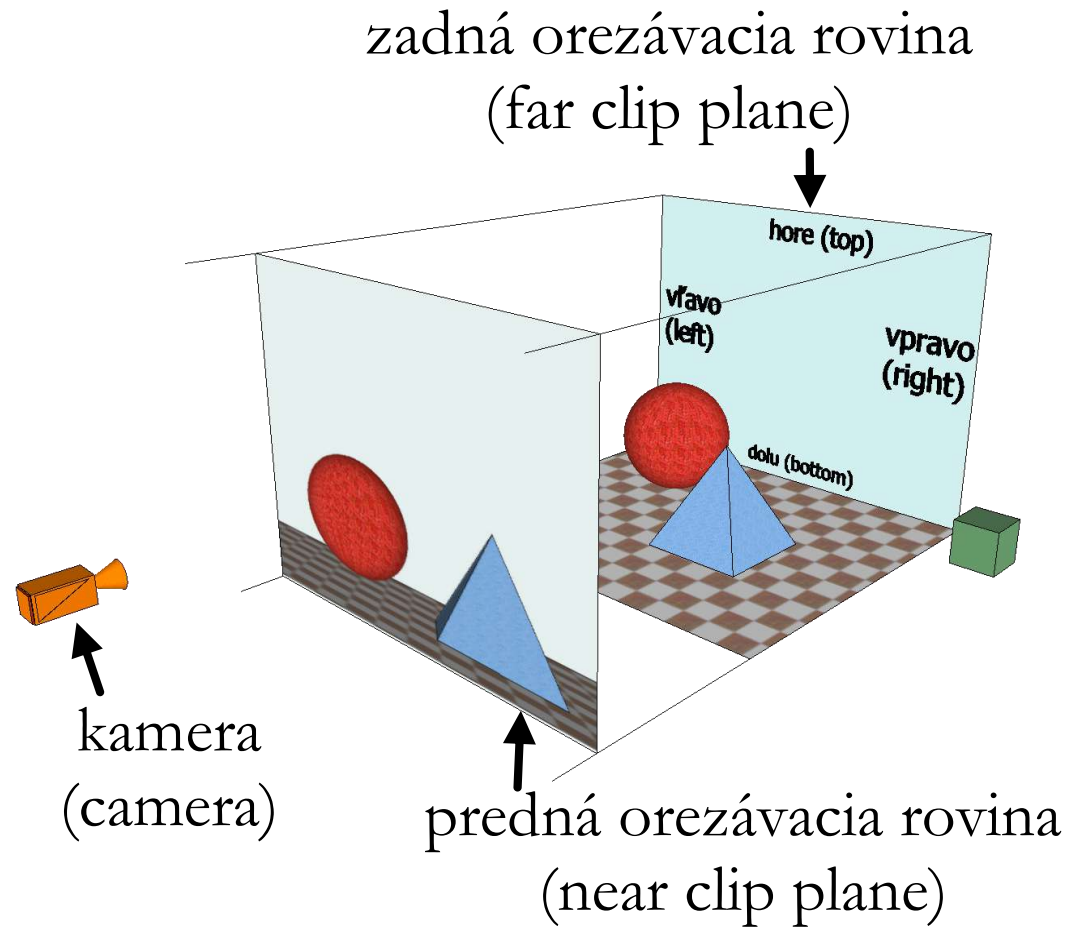
TYPY PROJEKČNÝCH TRANSFORMÁCIÍ (3D→2D)

- Ortografická (kolmá) projekcia (rovnobežné lúče, rozmery objektu sa nezmenšujú s hĺbkou)
 - Mongeova projekcia – nárys, pôdorys, bokorys.
 - Axonometria:
 - Izometria (všetky osi pod rovnakým uhlom, mierka rovnaká).
 - Dimetria (dve osi majú rovnaké skreslenie).
 - Trimetria (každá os má iné skreslenie).
 - Technická axonometria (projekcia) (šikmá, jedna os v skutočnej mierke).
- Perspektíva (zbiehavé lúče, rozmery objektu sa zmenšujú s hĺbkou)
 - Jednobodová perspektíva (1 úbežník (bod, kam sa zbiehajú lúče)).
 - Dvojbodová perspektíva (2 úbežníky).
 - Trojbodová perspektíva (3 úbežníky).
 - Panoramatická perspektíva (pre veľké uhly pohľadu).
- Špeciálne projekcie
 - Sférická projekcia (mapy, XR (VR+AR+MR)).
 - Cylindrická projekcia (mapy, panorámy).
 - Konická projekcia (mapy).



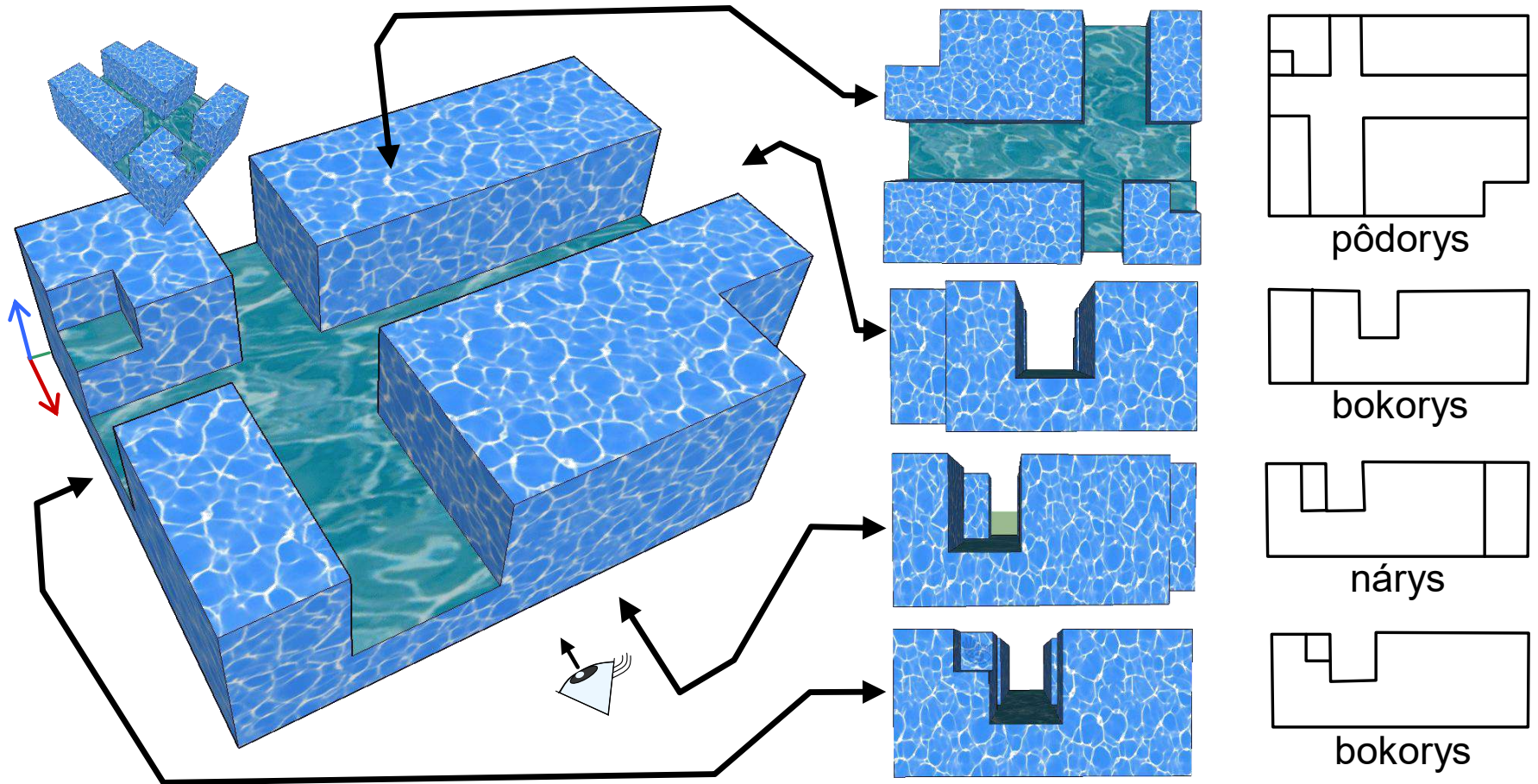
ORTOGRAFICKÁ (KOLMÁ) PROJEKCIA

ORTOGRAFICKÁ KAMERA



Ortografická projekcia

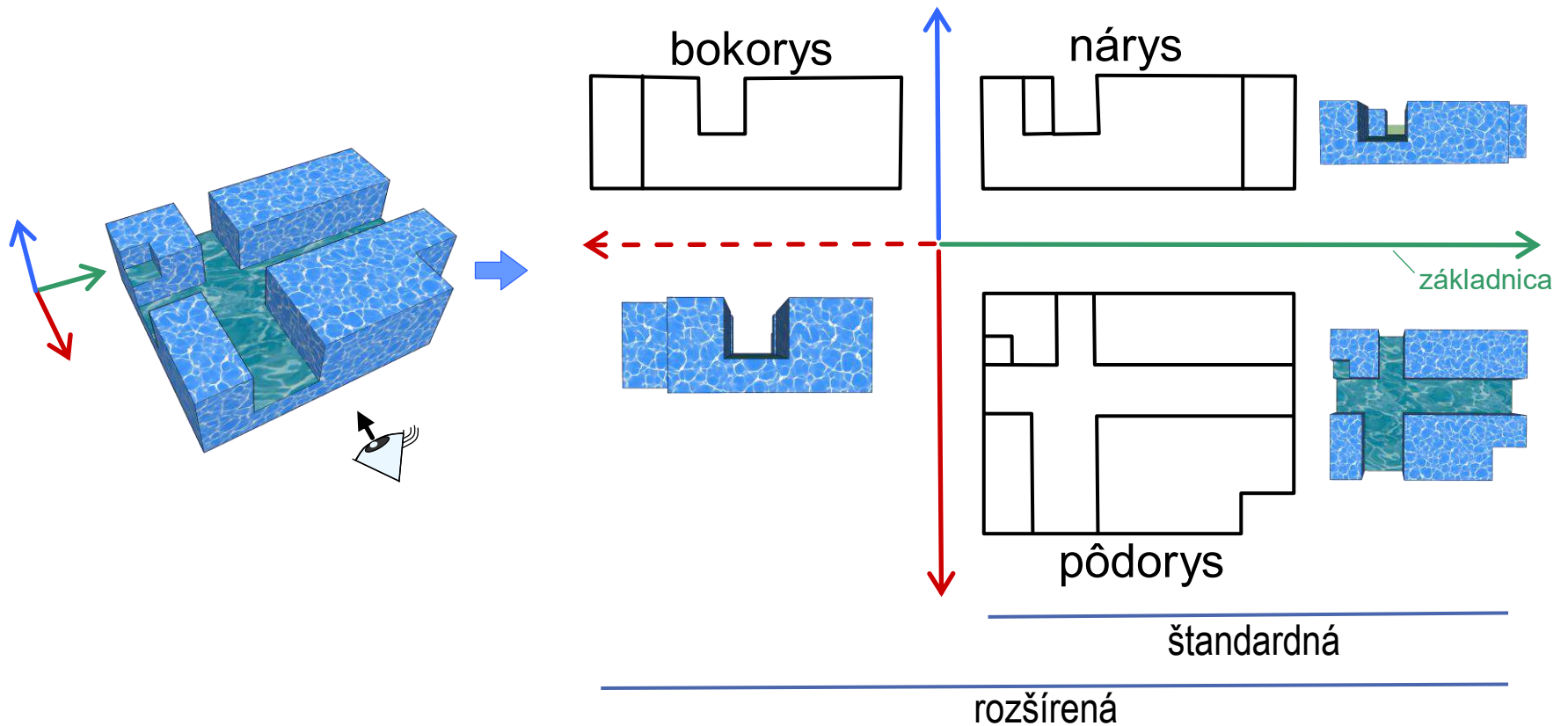
ORTOGRAFICKÁ (KOLMÁ) PROJEKCIA



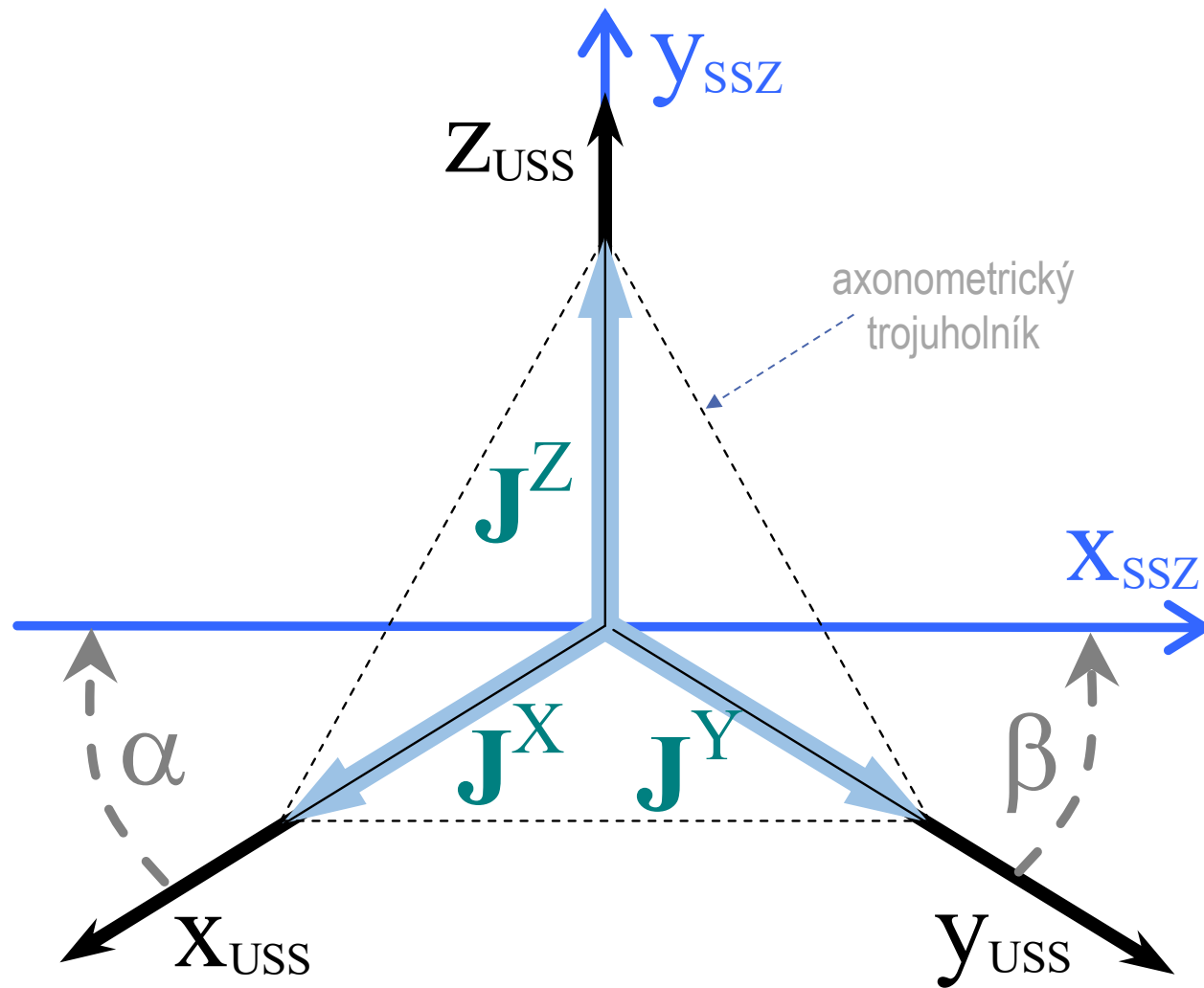
MONGEOVA PROJEKCIA



Gaspard Monge
(1746-1818)



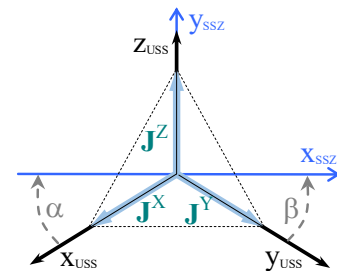
AXONOMETRIA



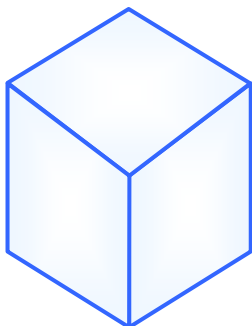
AXONOMETRIA

$$x_{SSZ} = -\mathbf{J}^X \cdot \cos(\alpha) \cdot x_{USS} + \mathbf{J}^Y \cdot \cos(\beta) \cdot y_{USS}$$

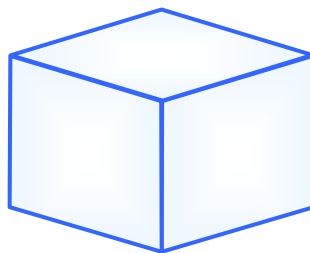
$$y_{SSZ} = -\mathbf{J}^X \cdot \sin(\alpha) \cdot x_{USS} - \mathbf{J}^Y \cdot \sin(\beta) \cdot y_{USS} + \mathbf{J}^Z \cdot z_{USS}$$



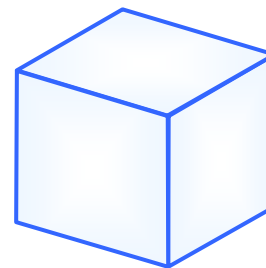
- izometria: $\mathbf{J}^X = \mathbf{J}^Y = \mathbf{J}^Z$ a $\alpha = \beta$
- dimetria: $\mathbf{J}^X = \mathbf{J}^Y$ a $\alpha = \beta$
- trimetria: $\mathbf{J}^X \neq \mathbf{J}^Y \neq \mathbf{J}^Z$ a $\alpha \neq \beta$
- technická axonometria: $\mathbf{J}^X = \mathbf{J}^Y$, $\mathbf{J}^Z = \frac{1}{2} \cdot \mathbf{J}^X$, $\alpha = 45^\circ$, $\beta = 0^\circ$



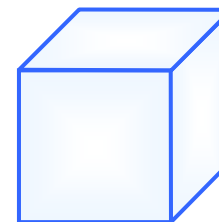
Izometria



Dimetria



Trimetria



Technická
axonometria

PERSPEKTÍVA

HISTORICKÉ NÁVÄZNOSTI



Leonardo da Vinci – použil matematické princípy lineárnej perspektívy (rovnobežky, horizont a úbežník) na vytvorenie ilúzie hĺbky na rovnom povrchu. O perspektívnom zobrazovaní hovorí (da Vinciho okno): *„Perspektíva nie je nič iné ako, videnie miesta za oknom z číreho skla, ako plochu, na ktorú sú zobrazované predmety spoza skla”*

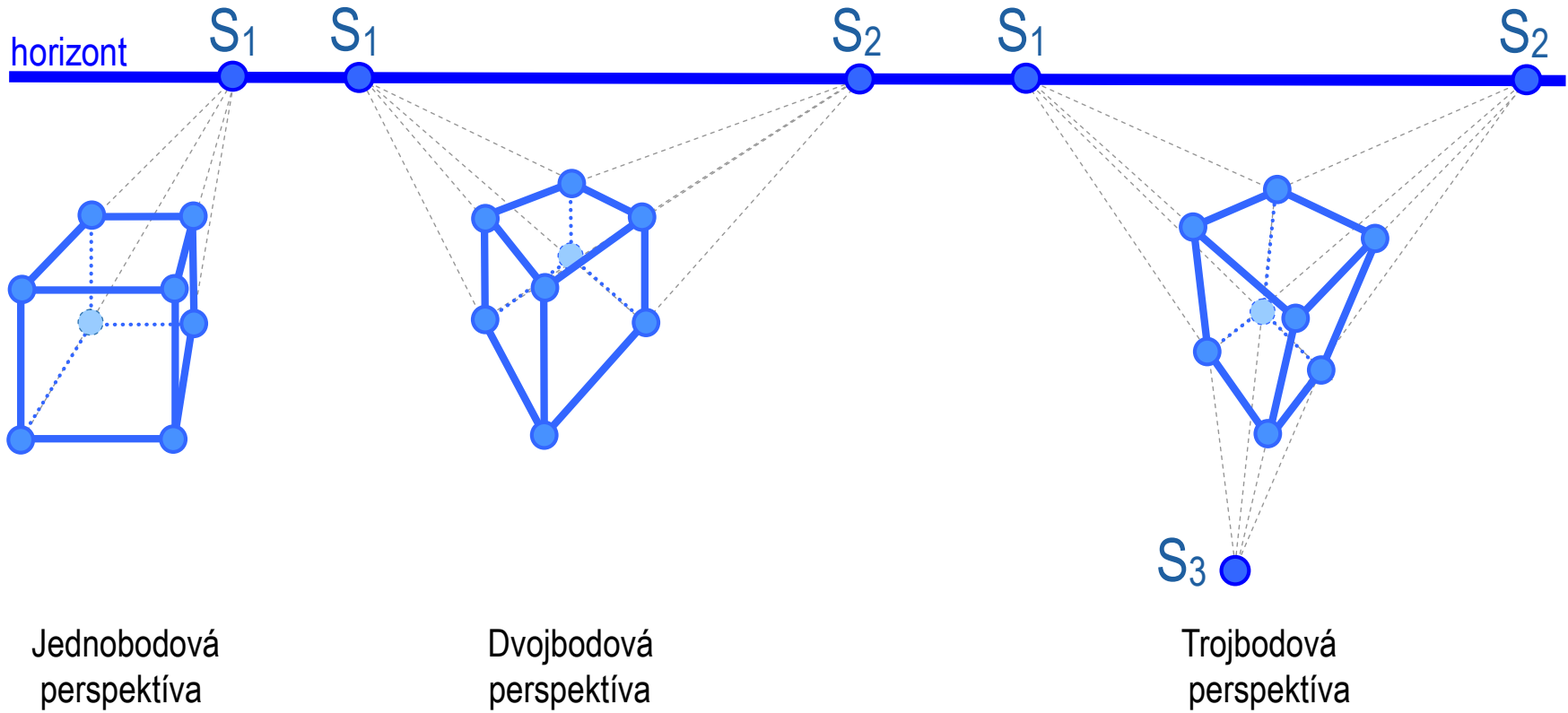
(„Perspective is nothing else than seeing a place or objects behind a plate of glass, quite transparent, on the surface of which the objects behind the glass are to be drawn“)



PERSPEKTÍVA

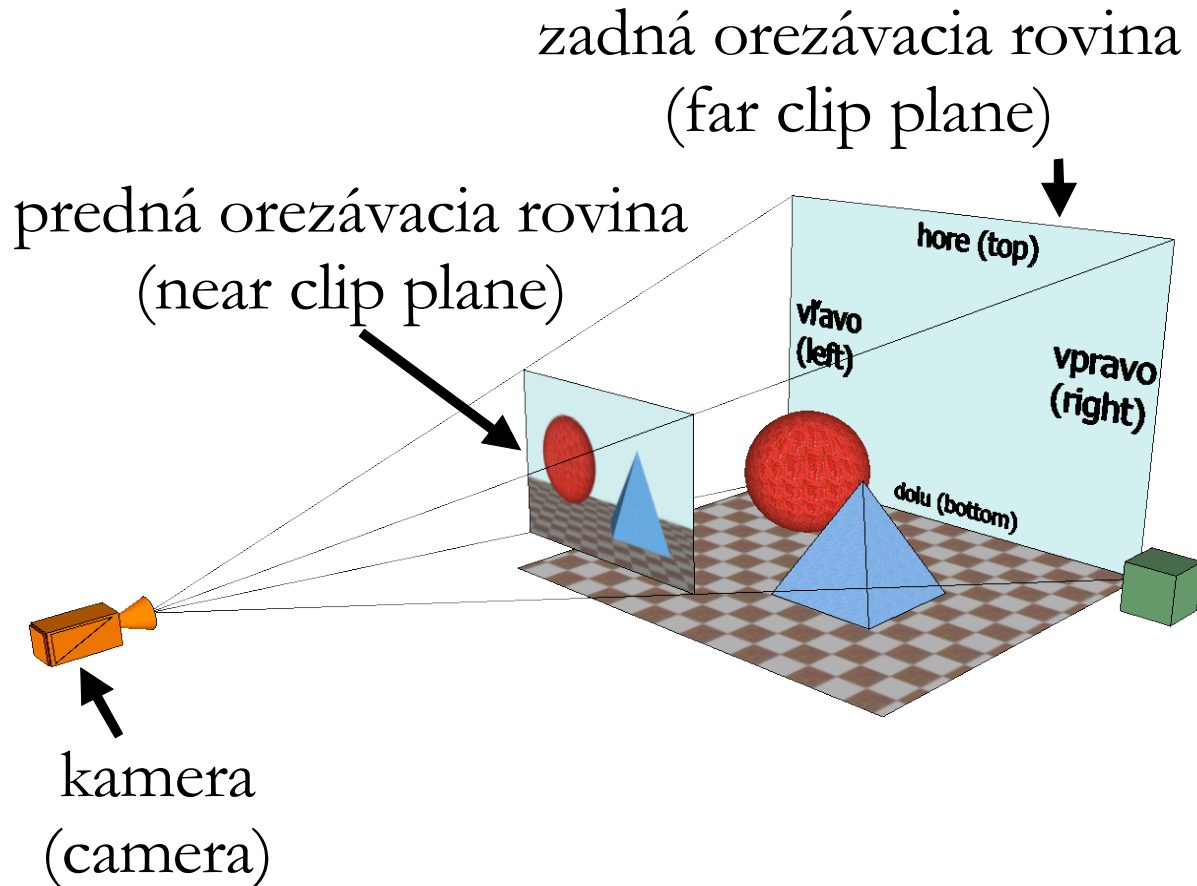
TYPY

S_n - úbežník



PERSPEKTÍVA

PERSPEKTÍVNA KAMERA



Perspektívna projekcia

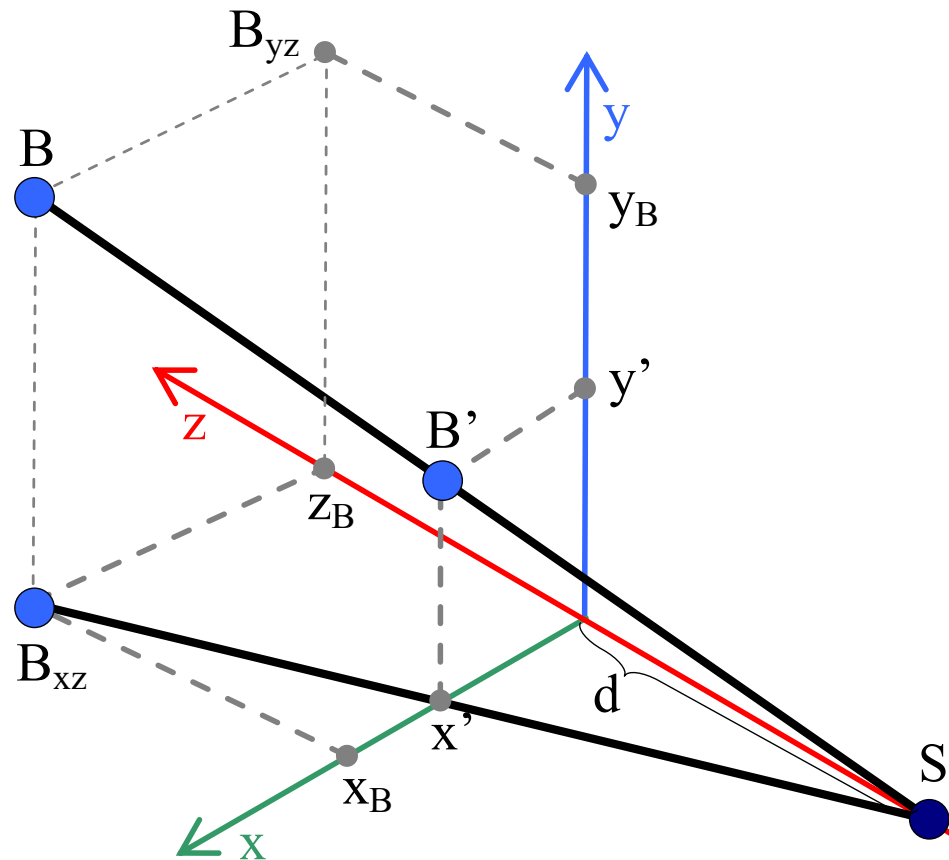
PERSPEKTÍVA

$$P = \frac{d}{d + z_B}$$

$$\frac{x'}{x_B} = \frac{y'}{y_B} = P$$

$$x' = P \cdot x_B$$

$$y' = P \cdot y_B$$



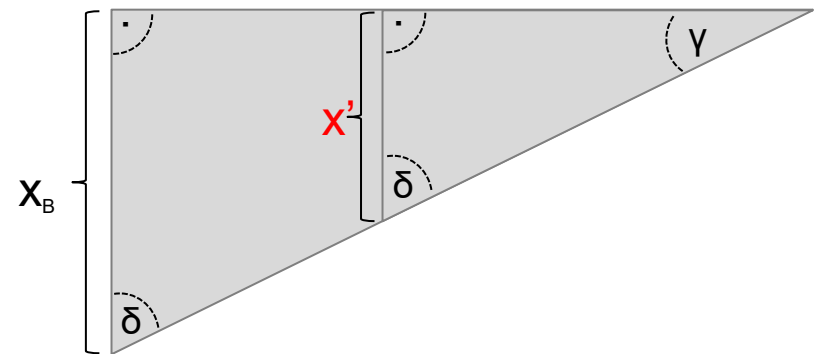
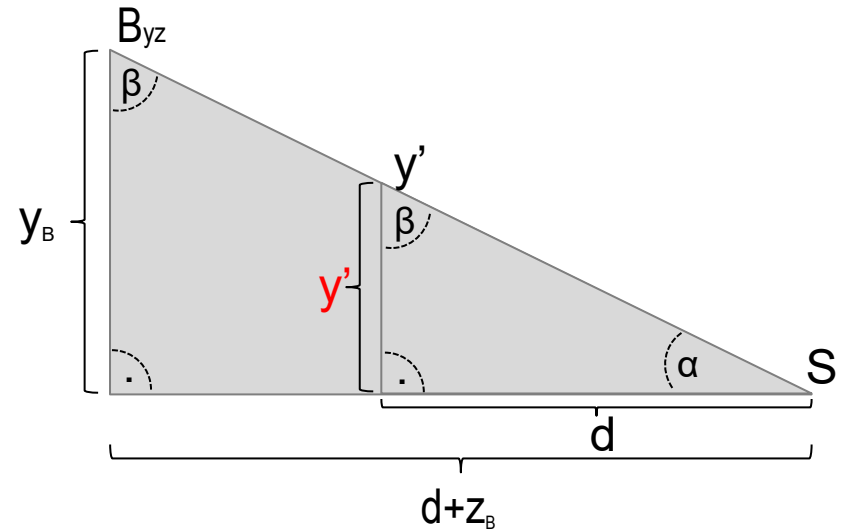
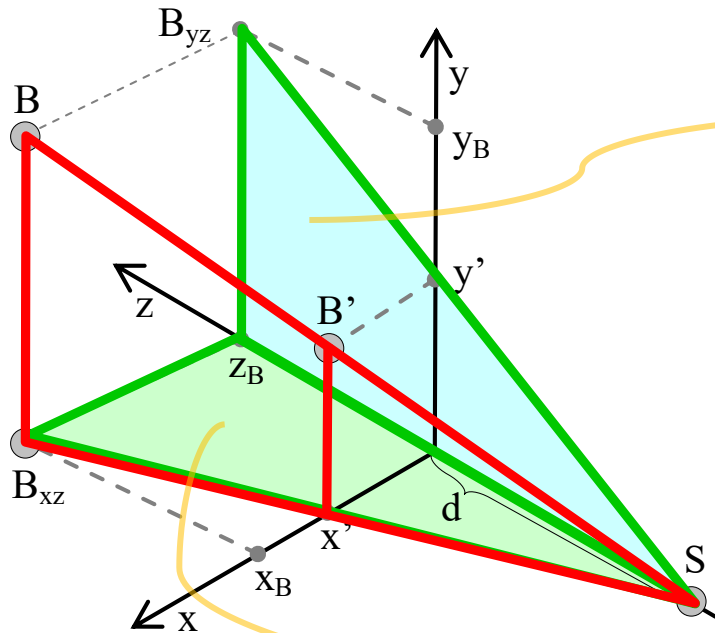
PERSPEKTÍVA

$$P = \frac{d}{d + z_B}$$

$$\frac{x'}{x_B} = \frac{y'}{y_B} = P$$

$$x' = P \cdot x_B$$

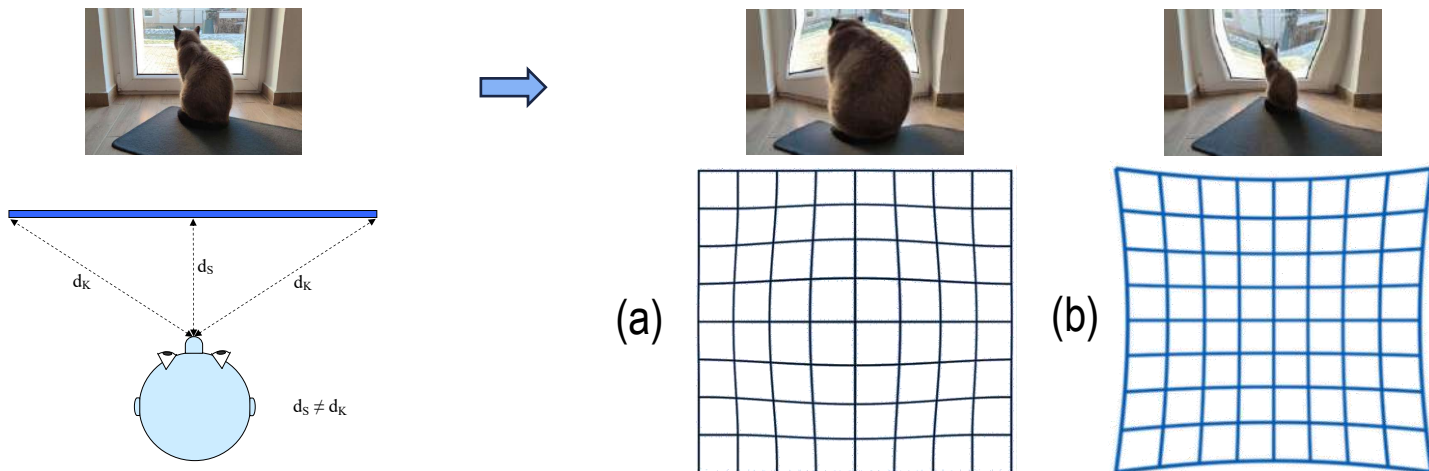
$$y' = P \cdot y_B$$



DISTORZIA OBRAZU (NELINEÁRNA TRANSFORMÁCIA)

Distorzia (skreslenie) obrazu – je vlastne optická chyba spôsobená tým, že priečne zväčšenie nie je po celej ploche obrazu rovnaké (vzdialenosť v strede je iná ako na kraji, v optike závisí aj od konštrukcie šošovky/rozptylky). Tým je narušená geometrická predmetová a obrazová podobnosť. Neskresľuje sa zobrazenie jednotlivých bodov, ale ich konfigurácia.

Pravouhlá matica lineárnych kriviek kolmá na optickú os kamery je vizuálne deformovaná na sieť konvexných kriviek ((a), eng. *barrel distorsion*) pri zväčšovaní vzdialenosti od osi kamery (konkávnych kriviek ((b) eng. *pincushion distorsion*) pri zmenšovaní vzdialenosti).



RYBIE OKO (NELINEÁRNA TRANSFORMÁCIA)

Zobrazenia, ktoré vzniknú na základe týchto rovníc nazývame:

- *ortografickým* (zobrazuje objekty ako keby boli premietnuté kolmo na rovinu snímача),

$$y = f \cdot \sin(\beta)$$

- *rovnoplochým (ekvivosolným)* (zachováva rovnakú plochu pre rovnaký uhol) a

$$y = 2 \cdot f \cdot \sin\left(\frac{\beta}{2}\right)$$

- *ekvidišťantným* zobrazením (uhol lúča je priamo úmerný vzdialenosti od stredu obrazu)

$$y = f \cdot \beta$$

kde:

y - vzdialenosť snímku svetelného lúča a prechádzajúceho bodom O, od hlavného stredového bodu S

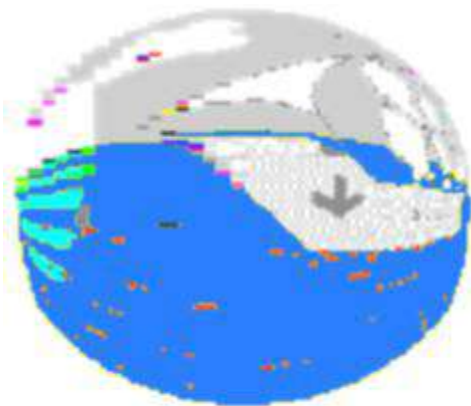
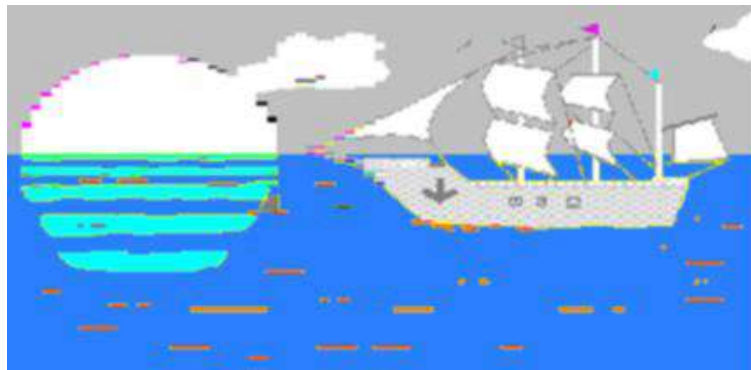
f - ohnisková vzdialenosť objektívu

β - uhol, ktorý lúč vytvára s optickou osou o objektívu.

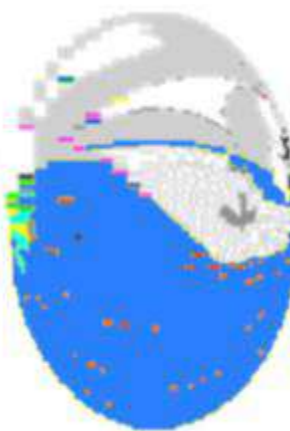
RYBIE OKO

PRÍKLADY

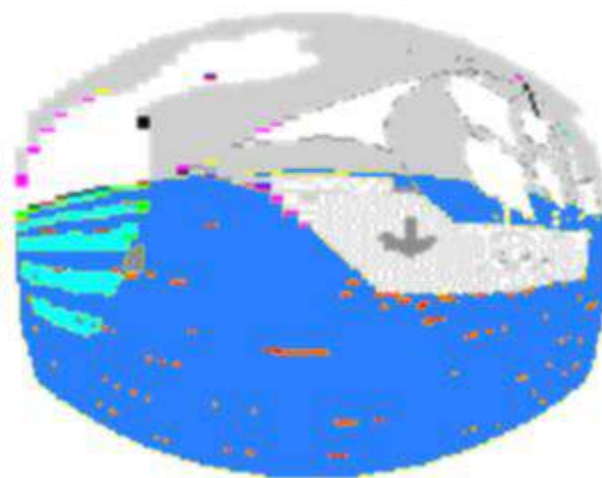
Originál



Ortografické



Rovnoploché



Ekvidištantné

RYBIE OKO

PRÍKLADY

Originál



Ortografické



Rovnoploché



Ekvidištantné

Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

KRIVKY A PLOCHY V POČÍTAČOVEJ GRAFIKE

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 06

© 2025

VRSTVY VIZUALIZAČNÉHO PROCESU

1. Definovanie/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



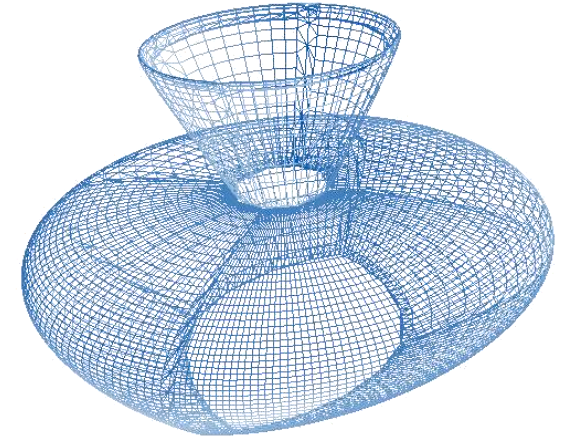
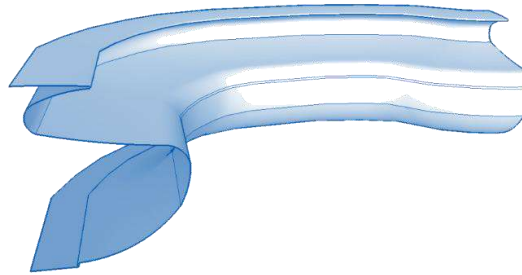
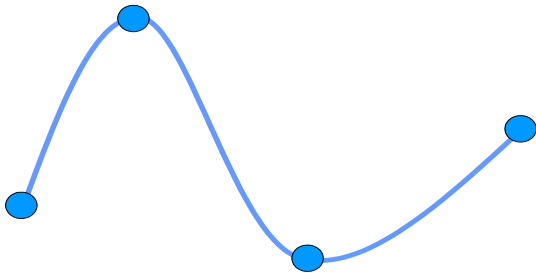
KRIVKY POUŽÍVANÉ V PG

- krivky dané analytickým popisom
- interpolačné krivky
- aproximačné krivky

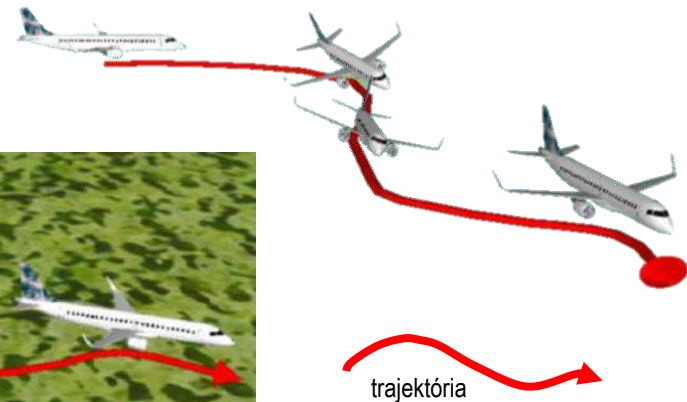
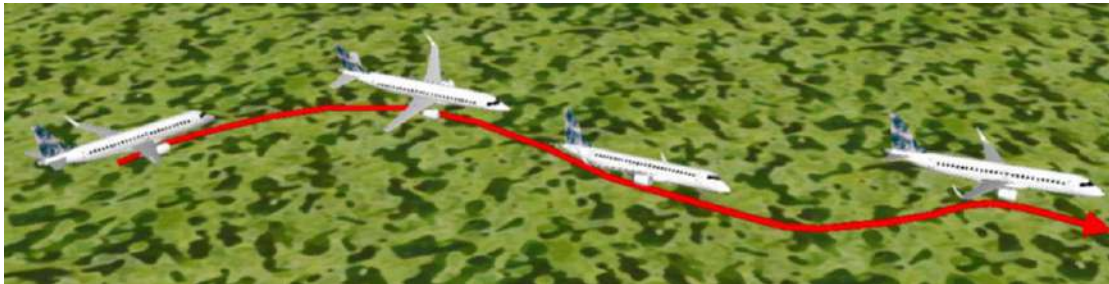


POUŽITIE KRIVIEK A PLÔCH V PG

- Geometrická reprezentácia



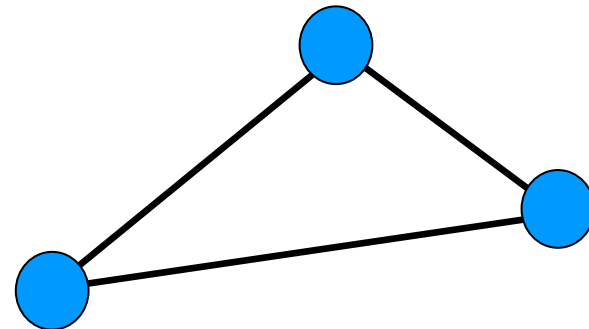
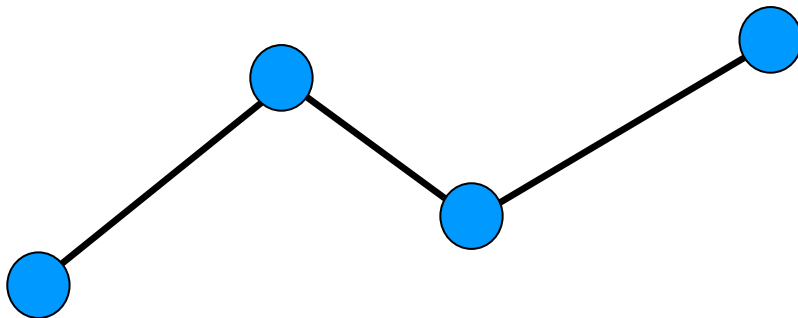
- Riadiaca funkcia



KRIVKOVÉ 1D ÚTVARY - TYPY

1D útvar definovaný vrcholmi a hranami (segmentami)-
polyline:

- neuzavretý (acyklický)
- uzavretý (cyklický)

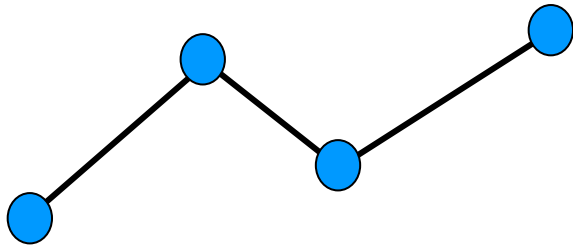


KRIVKOVÉ 1D ÚTVARY - TYPY

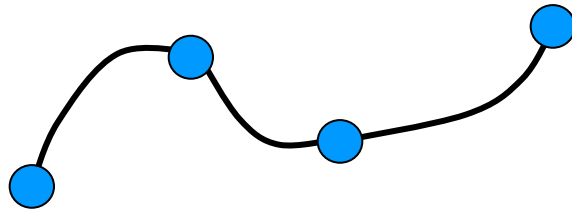
Podľa typov hrán (segmentov) krivky/čiary:

- lineárne
- nelineárne

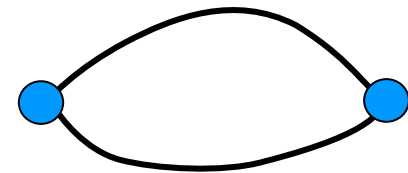
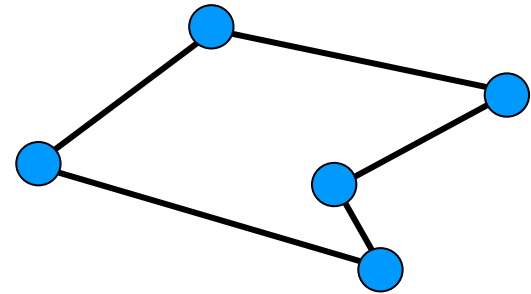
lineárne



nelineárne



acyklické
(neuzavreté)



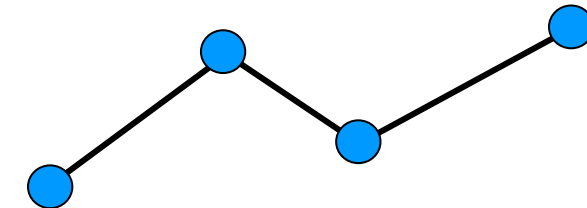
cyklické
(uzavreté)

KRIVKOVÉ 1D ÚTVARY - TYPY

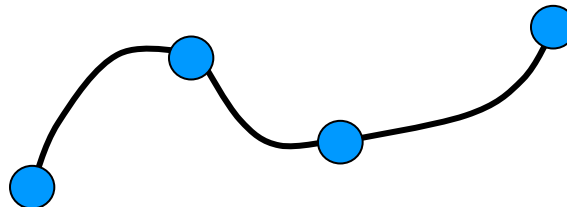
Podľa vplyvu vrcholov na tvar krivky/čiary:

- **Interpolačné** (vrcholy sú súčasťou krivky), typovo: *lineárna* a *nelineárna*.
- **Aproximačné** (vrcholy nemusia byť súčasťou krivky, ale vplývajú na jej tvar), typovo: *lineárna* a *nelineárna*.

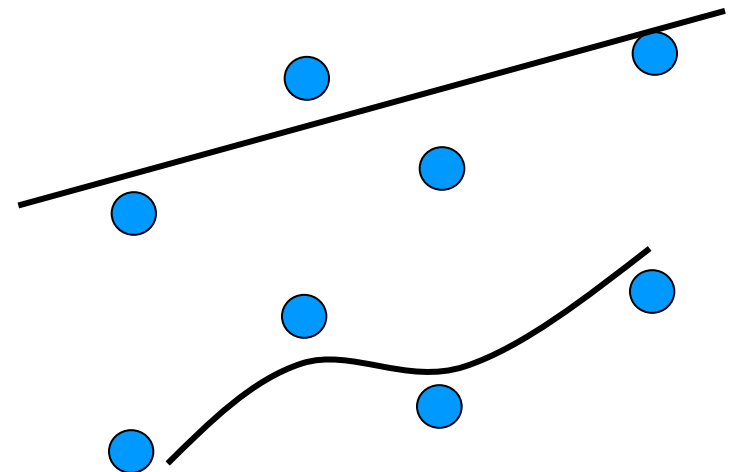
lineárne



nelineárne



interpolačné



aproximačné

POUŽÍVANÉ KRIVKY

- Lineárna interpolácia, lomená čiara
- Bézierove krivky stupňa n
- Racionálne Bézierove krivky stupňa n
- B-spline krivky stupňa n
- Kubické B-spline krivky
 - Uniformné
 - Neuniformné
- Racionálne B-spline krivky stupňa n
 - Uniformné
 - Neuniformné

} lineárne

} nelineárne

MODIFIKOVATEĽNOSŤ KRIVIEK

- Zmena polohy riadiacich vrcholov
- Zmena váh riadiacich vrcholov
- Modifikácia uzlového vektora



KRIVKY DEFINOVANÉ ANALYTICKY

$$x = f(t)$$

$$\left. \begin{array}{l} y = g(t) \end{array} \right\} t \in \langle zac, kon \rangle$$

$$z = h(t)$$

kde t je parameter funkcie (najčastejšie $\langle 0, 1 \rangle$)

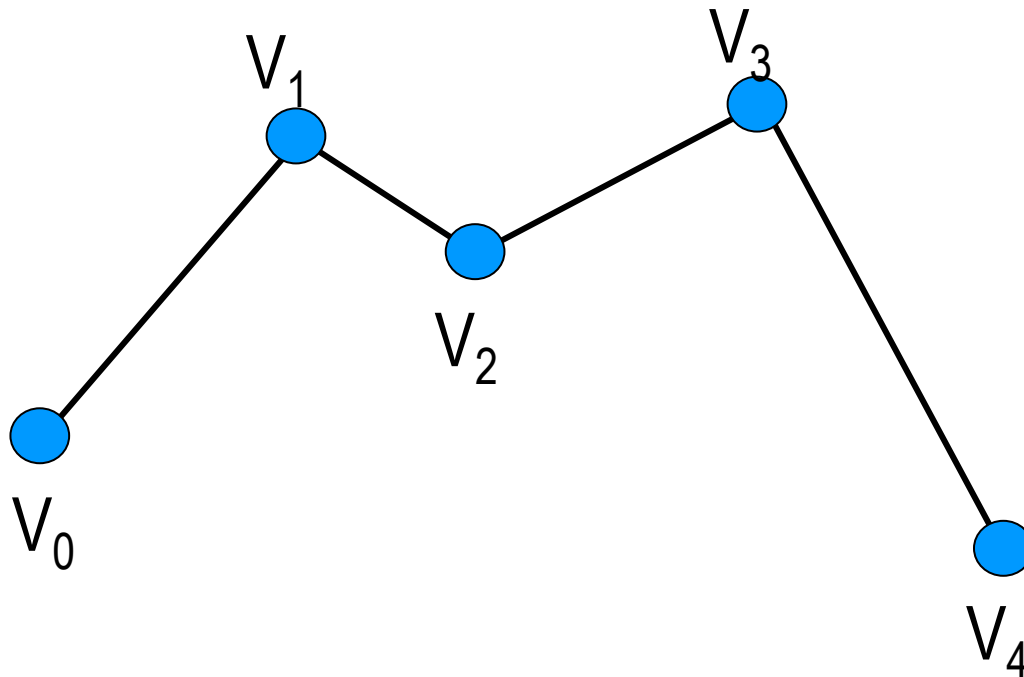
alebo (pre 2D)

$$y = f(x) \quad x \in \langle zac_x, kon_x \rangle$$

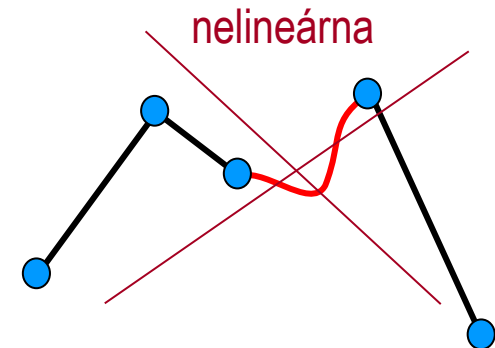
(pre 3D)

$$z = f(x, y) \quad x \in \langle zac_x, kon_x \rangle, y \in \langle zac_y, kon_y \rangle$$

LINEÁRNA INTERPOLÁCIA, LOMENÁ ČIARA

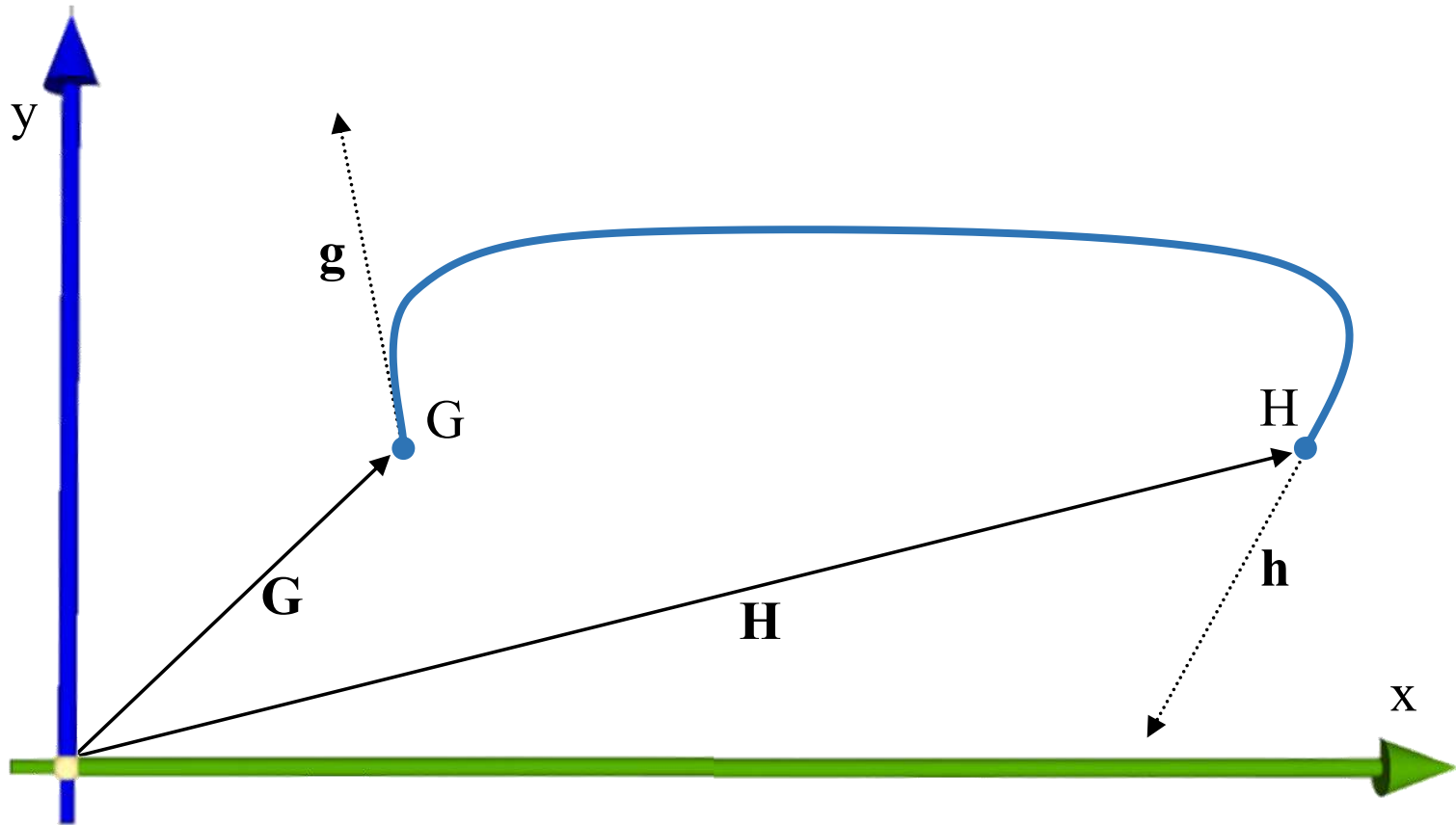


všetky segmenty musia byť lineárne



FERGUSONOVA KRIVKA (JAMES C. FERGUSON, 1963, BOEING)

Fergusonova krivka (interpolačná krivka)



FERGUSONOVA KRIVKA

$$\mathbf{P}(v) = \mathbf{m} \cdot v^3 + \mathbf{n} \cdot v^2 + \mathbf{p} \cdot v + \mathbf{q}$$

$v \in \langle 0, 1 \rangle$

$$\mathbf{P}_{xyz}(v) = \mathbf{m} \cdot v^3 + \mathbf{n} \cdot v^2 + \mathbf{p} \cdot v + \mathbf{q}$$

$$\mathbf{m} = 2 \cdot \mathbf{G}_{xyz} - 2 \cdot \mathbf{H}_{xyz} + \mathbf{g}_{xyz} + \mathbf{h}_{xyz}$$

$$\mathbf{n} = -3 \cdot \mathbf{G}_{xyz} + 3 \cdot \mathbf{H}_{xyz} - 2 \cdot \mathbf{g}_{xyz} - \mathbf{h}_{xyz}$$

$$\mathbf{p} = \mathbf{g}_{xyz}$$

$$\mathbf{q} = \mathbf{G}_{xyz}$$

FERGUSONOVA KRIVKA

$$\mathbf{P}(v) = A(v) \cdot \mathbf{G} + B(v) \cdot \mathbf{H} + C(v) \cdot \mathbf{g} + D(v) \cdot \mathbf{h}$$

$$\mathbf{G}, \mathbf{H}, \mathbf{g}, \mathbf{h} \in E^3(E^2), \quad v \in \langle 0, 1 \rangle$$

$$\mathbf{P}_{xyz}(v) = A(v) \cdot \mathbf{G}_{xyz} + B(v) \cdot \mathbf{H}_{xyz} + C(v) \cdot \mathbf{g}_{xyz} + D(v) \cdot \mathbf{h}_{xyz}$$

$$A(v) = 2 \cdot v^3 - 3 \cdot v^2 + 1$$

$$B(v) = -2 \cdot v^3 + 3 \cdot v^2$$

$$C(v) = v^3 - 2 \cdot v^2 + v$$

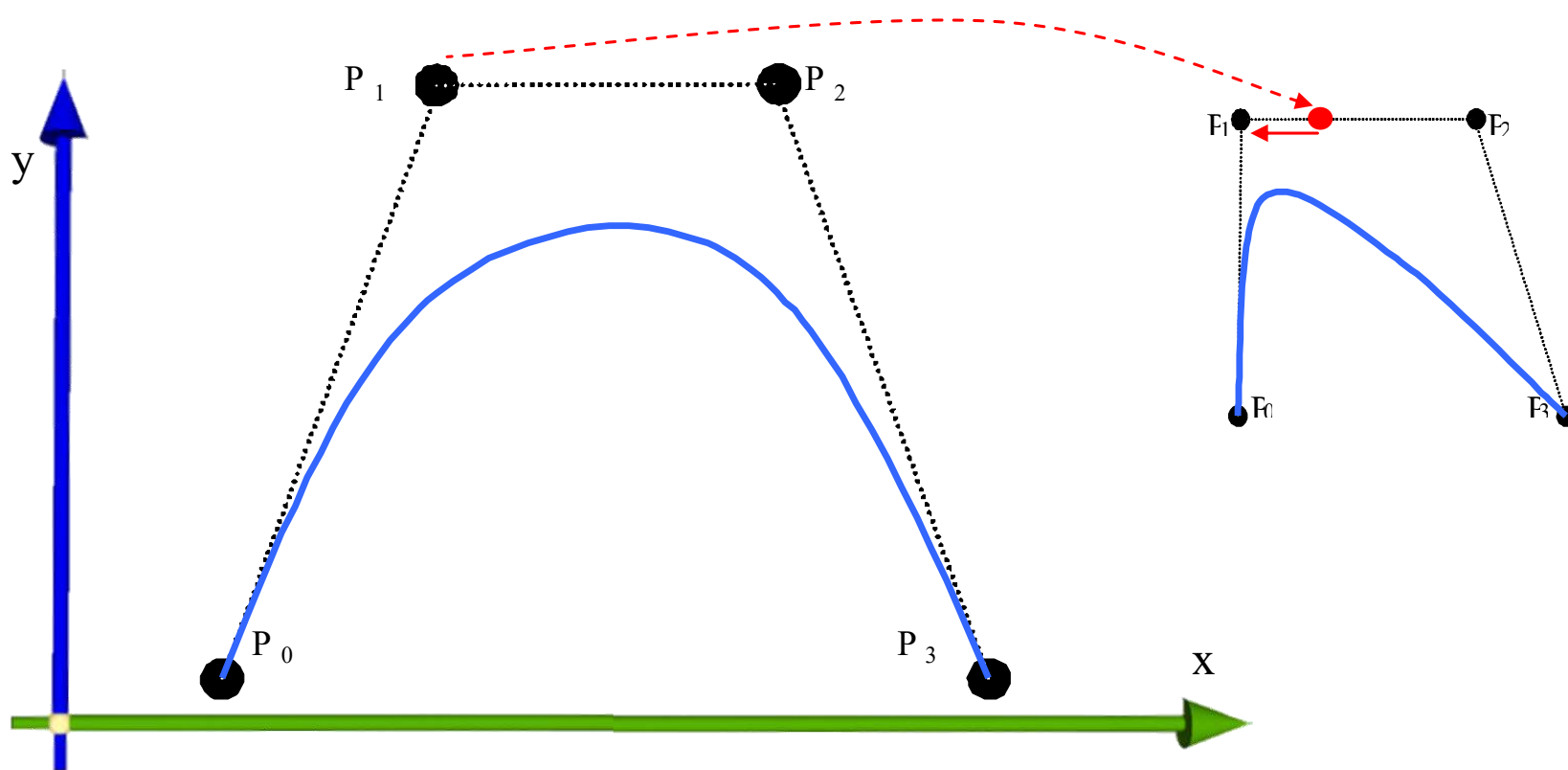
$$D(v) = v^3 - v^2$$



Hermitove polynómy
(Ch.Hermit, 1869)

BÉZIEROVA KRIVKA (PIERRE BÉZIER, 1960's, RENAULT)

Bézierova kubická krivka (aproximačná krivka)

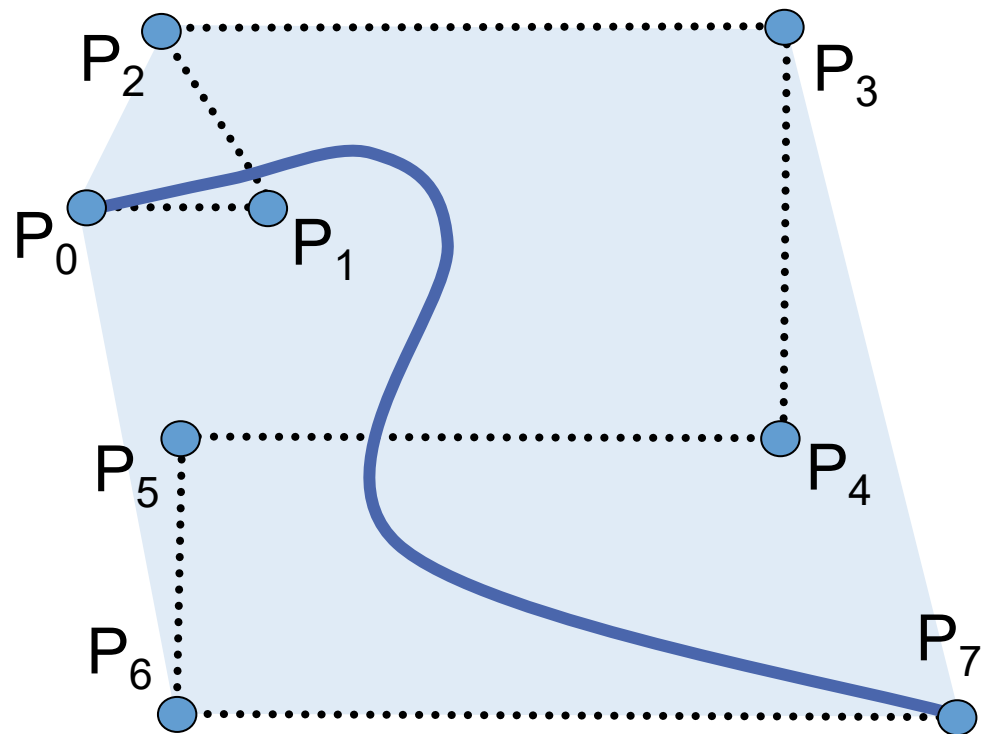


VLASTNOSTI BÉZIEROVEJ KRIVKY

Základné vlastnosti Bézierovej krivky stupňa n :

- je aproximačného typu ale interpoluje koncové vrcholy, je uniformná
- Definovaná polynomiálnou funkciou stupňa n , kde

$$n = \text{počet radiacich vrcholov} - 1$$
- Leží v konvexnom obale radiacich vrcholov
- Pseudolokálna kontrola
- Afinná invariancia (po aplikovaní afinných transformácií zachováva tvar)



BÉZIEROVA KRIVKA STUPŇA n

Ide o aproximačnú krivku, ktorá je polynómom n -tého stupňa:

$$\mathbf{R}(t) = \sum_{i=0}^m \mathbf{P}_i \cdot Be_{im}(t)$$

Bernsteinove polynómy
(S.N.Bernstein, 1912)



$$\mathbf{P}_i \in E^3(E^2)$$

$$Be_{im}(t) = \sum_{t \in \langle 0,1 \rangle} \binom{m}{i} \cdot t^i \cdot (1-t)^{m-i}$$

BÉZIEROVA KUBICKÁ KRIVKA

$$\mathbf{P}(t) = \mathbf{P}_0 \cdot B_0(t) + \mathbf{P}_1 \cdot B_1(t) + \mathbf{P}_2 \cdot B_2(t) + \mathbf{P}_3 \cdot B_3(t)$$

$$\mathbf{P}_i \in E^3(E^2), \quad t \in \langle 0, 1 \rangle$$

$$\mathbf{P}_{xyz}(t) = \mathbf{P}_{0xyz} \cdot B_0(t) + \mathbf{P}_{1xyz} \cdot B_1(t) + \mathbf{P}_{2xyz} \cdot B_2(t) + \mathbf{P}_{3xyz} \cdot B_3(t)$$

$$\left. \begin{aligned} B_0(t) &= (1-t)^3 \\ B_1(t) &= 3 \cdot t \cdot (1-t)^2 \\ B_2(t) &= 3 \cdot t^2 (1-t) \\ B_3(t) &= t^3 \end{aligned} \right\} \text{ pre } t \in \langle 0, 1 \rangle$$

SPLINE KRIVKA

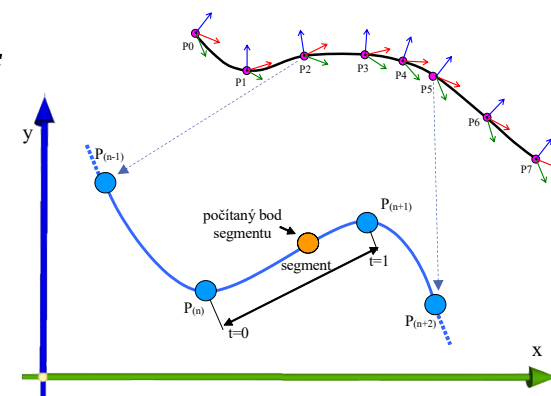
Spline funkciou stupňa m pre daných $n+1$ bodov $X_i = (x_i, y_i)$, $i = 0..n$, $x_0 < x_1 < \dots < x_n$, nazývame funkciu $f(x)$, pre ktorú na intervale $\langle x_0, x_n \rangle$ platí:

- $f(x) = f_k(x)$ na intervale $\langle x_k, x_{k+1} \rangle$, kde f_k je polynóm stupňa m ,
- $f(x)$ má spojité derivácie $f(0), f(1), \dots, f(m-1)$.

Najčastejšie sa používajú kubické spline funkcie ($m = 3$)

CATMULL-ROM SPLINE KRIVKA (1974)

Patrí medzi interpolačné krivky. Krivka prechádza cez všetky riadiace body P_i . Neexistujú žiadne diskontinuity v smere a veľkosti dotyčnice (C^1) ale druhá derivácia je lineárne interpolovaná v rámci každého segmentu (nie je C^2), čo spôsobuje, že zakrivenie sa lineárne mení po dĺžke segmentu. Výsledné body segmentu krivky môžu ležať mimo domény $P_1 \rightarrow P_2$. Aj keď segment splinu je definovaný pomocou štyroch riadiacich bodov, spline môže mať ľubovoľný počet ďalších riadiacich bodov.



$$Q(t) = \frac{1}{2} \cdot (A \cdot t^0 + B \cdot t + C \cdot t^2 + D \cdot t^3), \quad t \in \langle 0, 1 \rangle$$

$$A = 2 \cdot P_{(n)xyz}$$

$$B = -P_{(n-1)xyz} + P_{(n+1)xyz}$$

$$C = 2 \cdot P_{(n-1)xyz} - 5 \cdot P_{(n)xyz} + 4 \cdot P_{(n+1)xyz} - P_{(n+2)xyz}$$

$$D = -P_{(n-1)xyz} + 3 \cdot P_{(n)xyz} - 3 \cdot P_{(n+1)xyz} + P_{(n+2)xyz}$$

$$P_i \in E^3(E^2)$$

B-SPLINE KRIVKA

B-spline krivky sú zovšeobecnením Bézierových kriviek, miesto Bernsteinových polynómov sa používajú jednoduchšie funkcie.



Coonsove polynómy
(S.A.Coons, 1966)

Kubický Coonsov B-spline

$$\mathbf{R}(t) = \sum_{i=0}^3 \mathbf{P}_i \cdot Co_i(t), \quad t \in \langle 0,1 \rangle$$

$P_i \in E^3(E^2)$

B-SPLINE KRIVKA

Kubický Coonsov B-spline

$$\mathbf{R}(t) = \sum_{i=0}^3 \mathbf{P}_i \cdot Co_i(t), \quad t \in \langle 0,1 \rangle$$

kde:

$\left\{ \begin{matrix} n \\ \mathbf{P}_i \\ i=0 \end{matrix} \right\}$ sú polohové vektory riadiacich vrcholov pre $i = 0-3$

$$Co_0(t) = -\frac{1}{6} \cdot t^3 + \frac{1}{2} \cdot t^2 - \frac{1}{2} \cdot t + \frac{1}{6}$$

$$Co_1(t) = \frac{1}{2} \cdot t^3 - t^2 + \frac{2}{3}$$

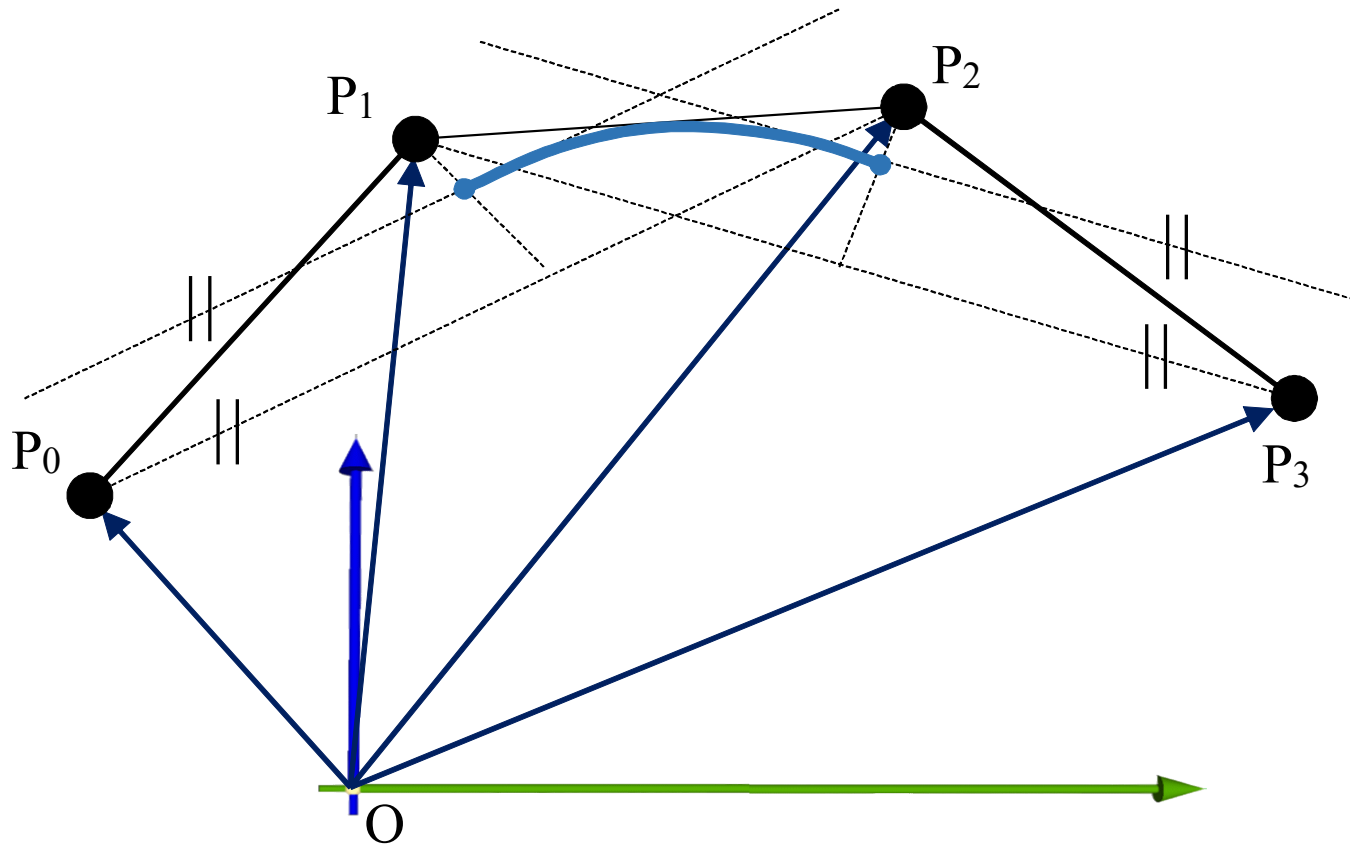
$$Co_2(t) = -\frac{1}{2} \cdot t^3 + \frac{1}{2} \cdot t^2 + \frac{1}{2} \cdot t + \frac{1}{6}$$

$$Co_3(t) = \frac{1}{6} \cdot t^3$$

pre $t \in \langle 0,1 \rangle$

B-SPLINE KRIVKA

Kubický Coonsov B-spline



VLASTNOSTI B-SPLINE KRIVKY

Základné vlastnosti B-spline krivky stupňa n :

- je aproximačná a je uniformná
- špeciálne zadané vstupné hodnoty (násobnosti) spôsobia, že krivka začína v bode P_0 a končí v bode P_{L+n-1} kde

n =počet riadiacich vrcholov

L =počet násobností riadiacich vrcholov

- Pseudolokálna kontrola a segmentovateľnosť
- Afinná invariancia (po aplikovaní afinných transformácií zachováva tvar)

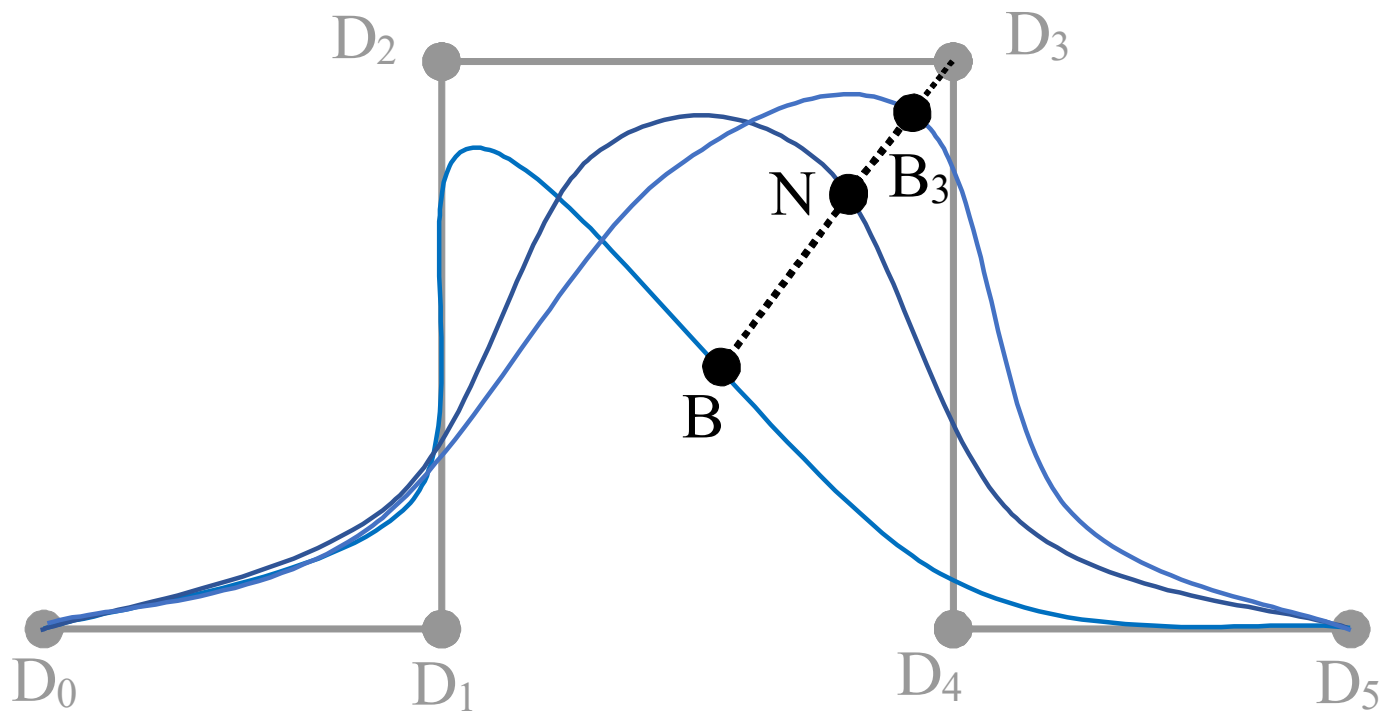
POROVNANIE BÉZIEROVEJ A B-SPLINE KRIVKY

Ak porovnáme Bézierové krivky a B-spline krivky pre daný polygón $\{P_i\}$, $i=0..n$, tak potom:

- Bézierova krivka je stupňa n , avšak B-spline krivka je zložená z $n-1$ segmentov (prvý segment je určený bodmi P_0, P_1, P_2, P_3 , druhý segment bodmi P_1, P_2, P_3, P_4 atď.), z ktorých každý je popísaný polynómom tretieho stupňa.
- Nevýhodou B-spline kriviek je, že krivka nezačína a nekončí v začiatočnom a koncovom bode polygónu. Túto nevýhodu možno odstrániť tak, že použijeme *náhradný polygón*, alebo sa zmení *násobnosť prvých a posledných prvkov uzlového vektora*.

RACIONÁLNE KRIVKY

Oproti reálnym pribúdajú váhy riadiacich vrcholov



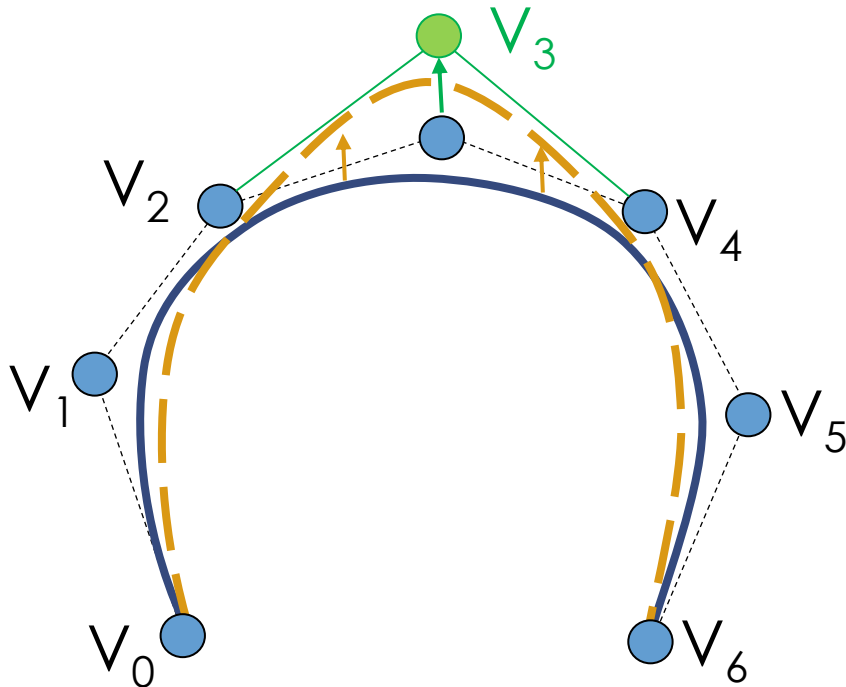
RACIONÁLNA BÉZIEROVA KRIVKA STUPŇA n

Oproti Beziérovej krivke pribúda ku každému riadiacemu vrcholu jeho váha (w_i):

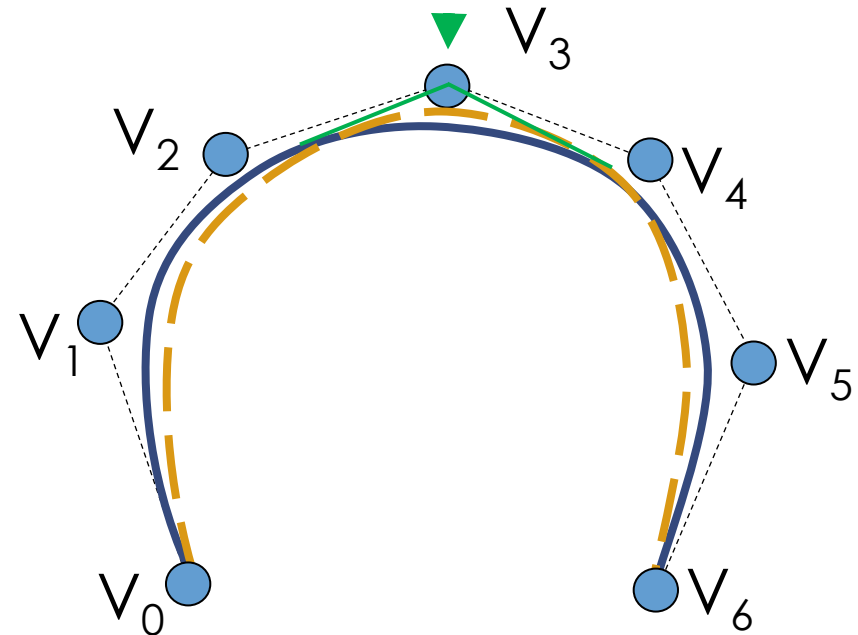
$$B^n(t) = \frac{\sum_{i=0}^n w_i V_i B_i^n(t)}{\sum_{i=0}^n w_i B_i^n(t)} \quad V_i \in E^3(E^2)$$

RACIONÁLNA BÉZIEROVA KRIVKA STUPŇA n

VPLYV ZMENY POLOHY A VÁHY



vplyv zmeny *polohy*
radiaceho bodu



vplyv zmeny *váhy*
radiaceho bodu

UNIFORMNOSŤ A NEUNIFORMNOSŤ

Voľba uzlovej postupnosti (napr $u_i = i$):

- **Uniformná** ak $u_{i+1} - u_i = \textit{konštanta}$ pre všetky i
- **Neuniformná** ľubovoľná neklesajúca postupnosť uzlov



NURBS – NonUNIFORM RATIONAL B-SPLINE

NURBS ako skratka pre neuniformný racionálny B-spline v sebe zahrňuje okrem toho, že je to B-spline krivka (či plocha) aj neuniformnosť a racionálnosť.

- *Neuniformnosť* znamená, že intervaly medzi jednotlivými uzlami nemusia byť rovnaké. Teda v konečnom dôsledku sa tým môže meniť celkový charakter krivky.
- *Racionálnosť* znamená, že každému riadiacemu vrcholu je priradená určitá váha, v ktorej je uložená miera toho, ako bude daný bod na krivku vplývať

VLASTNOSTI NURBS KRIVIEK (AJ PLÔCH)

- **Zovšeobecnenie** - NURBS je zovšeobecnením nižších kriviek (plôch) napr. Bézierova krivka (resp. plocha) a neracionálna B-splinova krivka je len špeciálnym prípadom NURBS krivky
- **Lokálnosť** - ak sa zmení váha riadiaceho bodu, alebo sa zmení jeho poloha, bude to mať lokálny účinok a ovplyvní sa tým len úsek krivky odpovedajúci príslušným $(p+1)$ úsekom uzlového vektora
- **Diferencovateľnosť** - vo vnútri svojho uzlového intervalu je racionálna bázická funkcia nekonečne veľa krát spojitou diferencovateľná, ak je menovateľ rôzny od nuly. V uzle je táto funkcia $(p-k)$ -krát spojitou diferencovateľná, pričom k je násobnosťou uzla a p je stupeň bázickej funkcie. Z tejto vlastnosti vyplýva fakt, že leží v konvexnom obale bodov $P_{i-p}, \dots, P_i$

NURBS - KRIVKA

NURBS krivka je vektorovo orientovaná polynomiálna funkcia v tejto forme:

$$\mathbf{C}(u) = \frac{\sum_{i=0}^n w_i V_i N_{i,p}(u)}{\sum_{i=0}^n w_i N_{i,p}(u)}$$

kde

- w_i sú váhy jednotlivých riadiacich bodov V_i
- $N_{i,p}(u)$ sú normalizované B-splinové (bázické) funkcie stupňa p
- u_i sú takzvané uzly tvoriace *uzlový vektor* (neuniformnosť určená násobnosťou uzlov), pričom uzly sú zoradené v neklesajúcej postupnosti $U = \{u_0, u_1, \dots, u_m\}$

PLOCHY POUŽÍVANÉ V PG

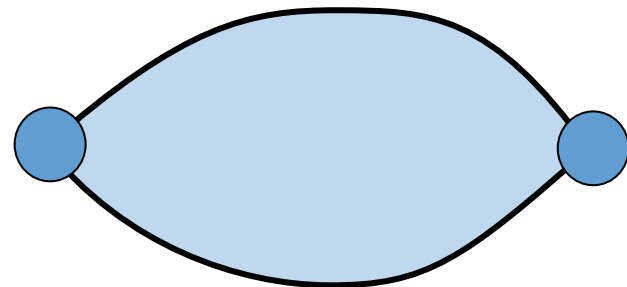
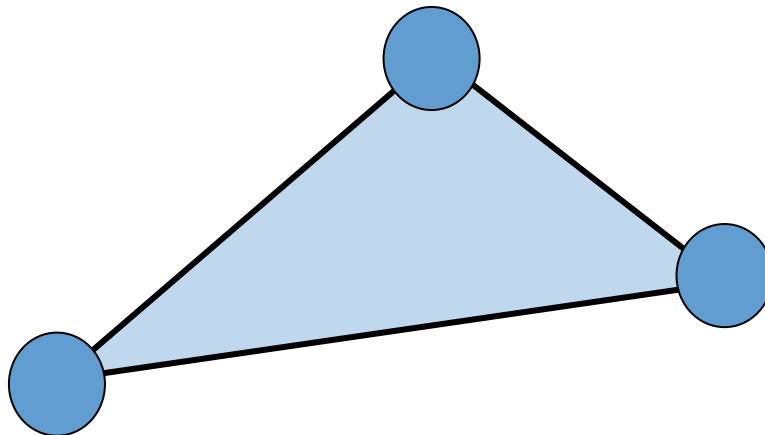
- Plochy dané analytickým popisom
- interpolačné plochy
- aproximačné plochy



PLOŠNÉ 2D ÚTVARY - TYPY

2D útvar definovaný vrcholmi a hranami - **polygón**:

- lineárny
- nelineárny



POUŽÍVANÉ PLOCHY

- Rovinné (pravítkové) plochy
- Bézierove plochy
- B-spline plochy
- Racionálne B-spline plochy
 - Uniformné
 - Neuniformné

} lineárne

} nelineárne

MODIFIKOVATEĽNOSŤ PLÔCH

- Zmena polohy riadiacich vrcholov
- Zmena váh riadiacich vrcholov
- Modifikácia uzlových vektorov



PLOCHY DEFINOVANÉ ANALYTICKY

$$x = f(t)$$

$$\left. \begin{array}{l} y = g(t) \\ z = h(t) \end{array} \right\} t \in \langle zac, kon \rangle$$

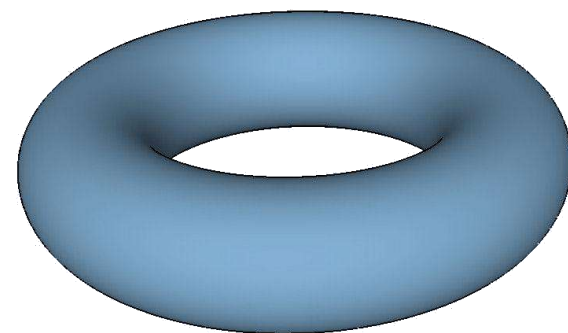
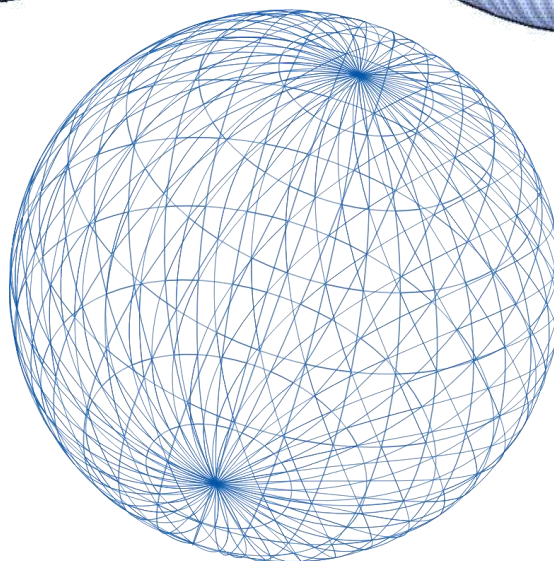
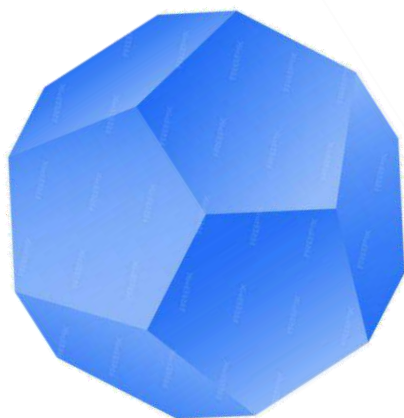
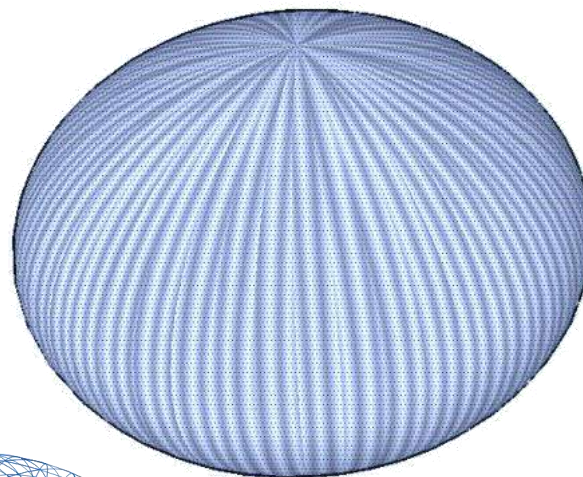
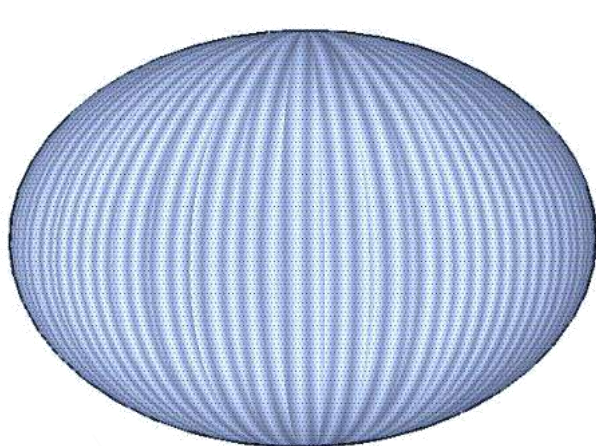
$$z = h(t)$$

kde t je parameter funkcie

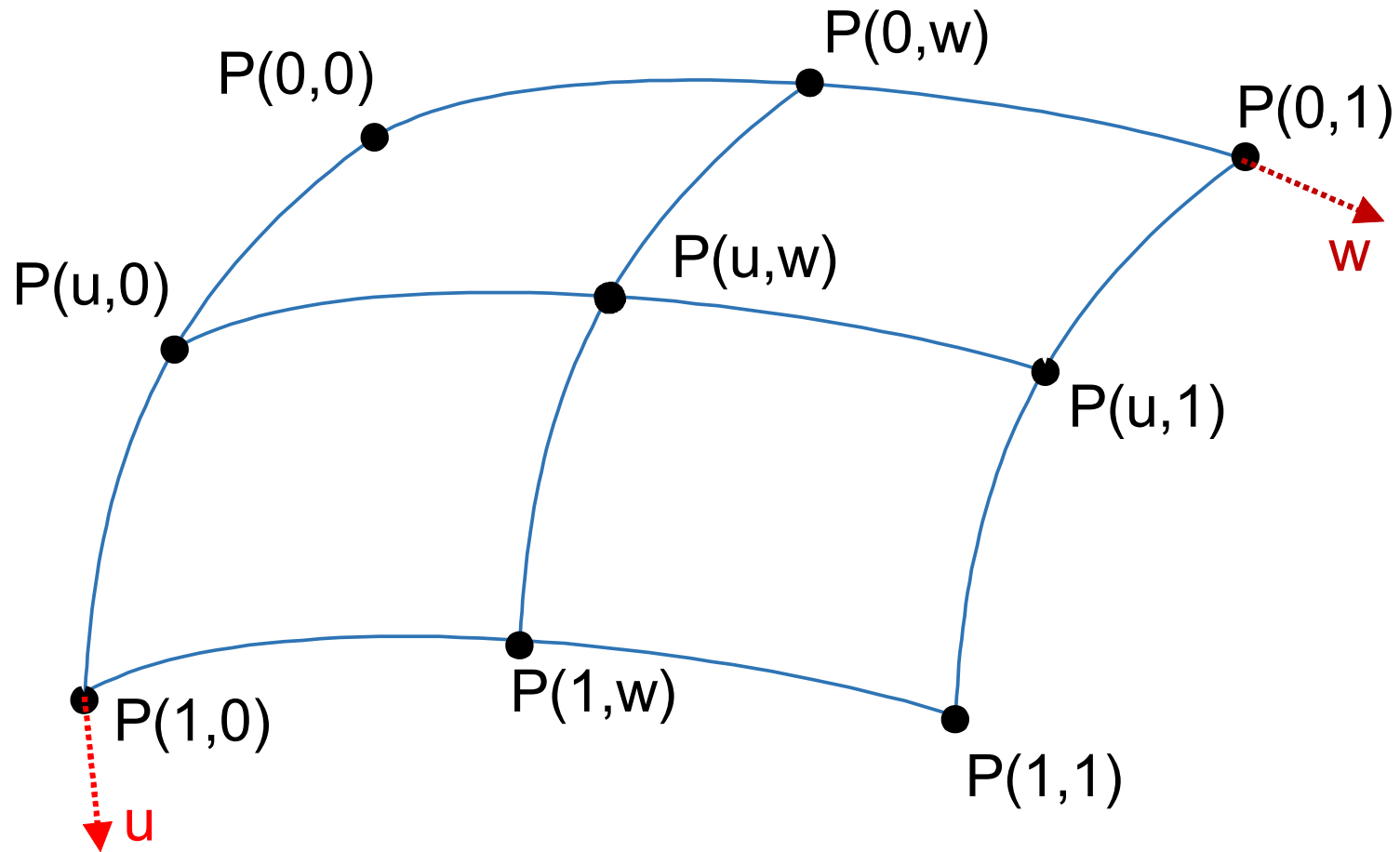
alebo

$$z = f(x, y) \quad x \in \langle zac_x, kon_x \rangle, y \in \langle zac_y, kon_y \rangle$$

PRÍKLADY PRAVIDELNÝCH ANALYTICKÝCH PLÔCH



BILINEÁRNA COONSOVA PLOCHA



BILINEÁRNA COONSOVA PLOCHA

Pokiaľ protiľahlé strany budú úsečky, dostaneme tzv. ***priamkovú*** (*pravítkovú*) plochu. Coonsova bilineárna plocha je teda všeobecnejšia ako plocha priamková.



BILINEÁRNA COONSOVA PLOCHA

$$\begin{bmatrix} 1-u & -1 & u \end{bmatrix} * \mathbf{M} * \begin{bmatrix} 1-w & -1 & w \end{bmatrix}^T = 0$$

kde:

u a w sú parametre ($u \in \langle 0, 1 \rangle$, $w \in \langle 0, 1 \rangle$), pretože bilineárny plát je určený okrajom $P(u, 0)$, $P(u, 1)$, $P(0, w)$ a $P(1, w)$

$$\mathbf{M} = \begin{bmatrix} \mathbf{P}_{xyz}(0,0) & \mathbf{P}_{xyz}(0,w) & \mathbf{P}_{xyz}(0,1) \\ \mathbf{P}_{xyz}(u,0) & \mathbf{P}_{xyz}(u,w) & \mathbf{P}_{xyz}(u,1) \\ \mathbf{P}_{xyz}(1,0) & \mathbf{P}_{xyz}(1,w) & \mathbf{P}_{xyz}(1,1) \end{bmatrix}$$

BILINEÁRNA COONSOVA PLOCHA

tvar sa prevedie na explicitný, čím dostaneme:

$$P(u, w) = [1 - u, u] * [P(0, w), P(1, w)]^T + [P(u, 0), P(u, 1)] * [1 - w, w]^T - [1 - u, u] * Q * [1 - w, w]$$

kde **Q** je matica:

$$Q = \begin{bmatrix} \mathbf{P}_{xyz}(0,0) & \mathbf{P}_{xyz}(0,1) \\ \mathbf{P}_{xyz}(1,0) & \mathbf{P}_{xyz}(1,1) \end{bmatrix}$$

BIKUBICKÁ COONSOVA PLOCHA

Je ďalším typom Coonsových plôch a predstavuje rozšírenie bilineárnej plochy, ale jej zadávanie a tvar je rovnaké. Rovnica popisujúca túto plochu je modifikáciou rovnice pre bilineárnu plochu.

$$\left[F_1(u), -1, F_2(u) \right] * \mathbf{M} * \left[F_1(w), -1, F_2(w) \right]^T = 0$$

kde $\mathbf{M}$ je matica ako u bilineárnej plochy a F_1, F_2 sú známe Fergusonove polynómy :

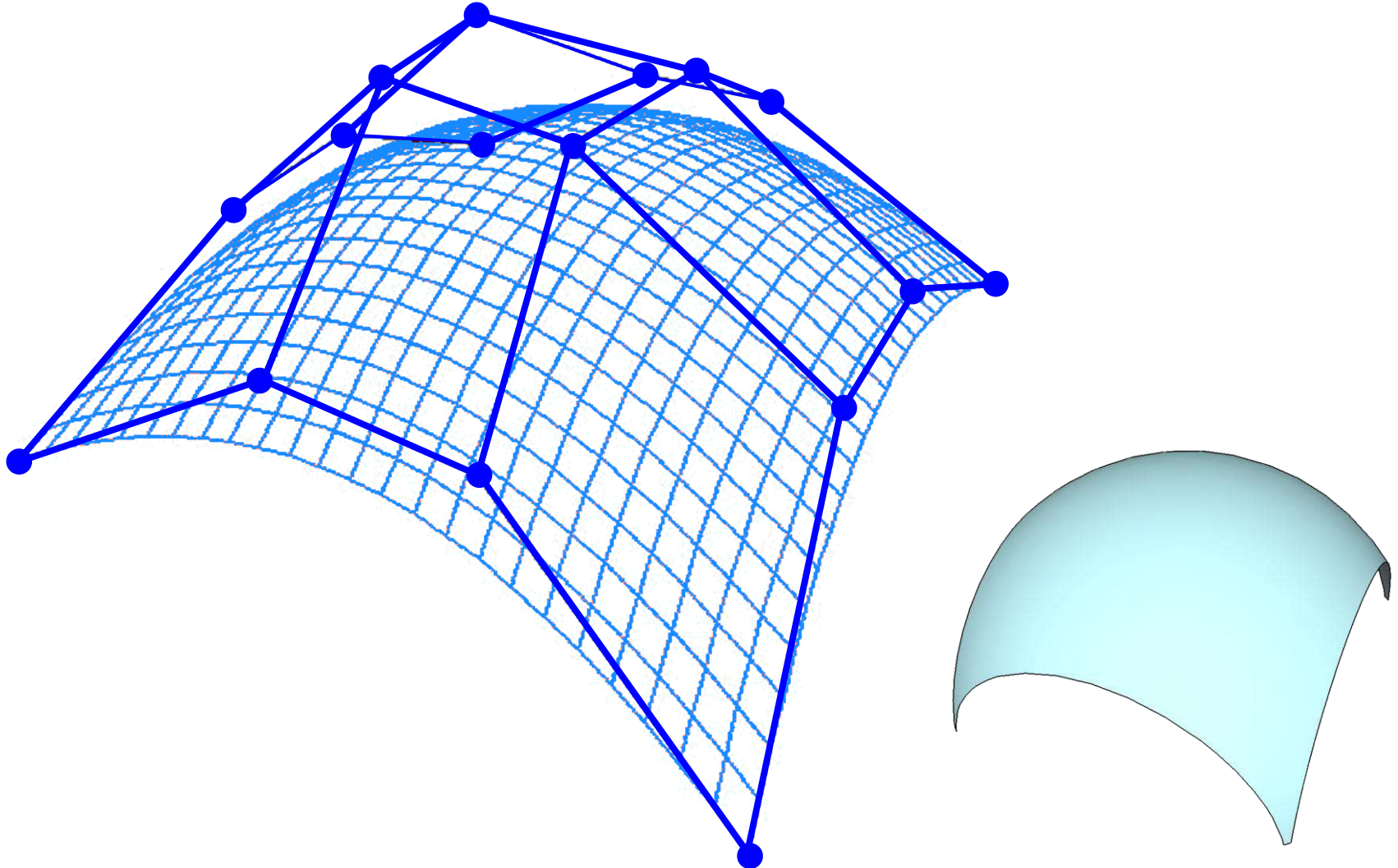
$$F_1(t) = 2 \cdot t^3 - 3 \cdot t^2 + 1$$

$$F_2(t) = -2 \cdot t^3 + 3 \cdot t^2$$

Pre lepšiu implementovateľnosť túto rovnicu upravíme a dostaneme analogickú explicitnú rovnicu ako v prípade bilineárnej plochy.

Základným problémom Coonsových plôch tohto typu je veľmi ťažké vyjadrenie priamych dotyčnicových vektorov, čím sa obtiažne dosahuje hladké spojenie dvoch Coonsových plátov.

BÉZIEROVA BIKUBICKÁ PLOCHA



BÉZIEROVA BIKUBICKÁ PLOCHA

Základná Bézierova plocha je nazývaná aj *Beziérová bikubická plocha* a je daná maticou 4×4 bodov, teda 16-imi uzlami B_{ij} , $i, j = 0, 1, 2, 3$. Plocha je následne definovaná explicitnou rovnicou:

$$z = [E(u), F(u), G(u), H(u)] * \mathbf{B} * [E(w), F(w), G(w), H(w)]^T$$

BÉZIEROVA BIKUBICKÁ PLOCHA

$$z = [E(u), F(u), G(u), H(u)] * \mathbf{B} * [E(w), F(w), G(w), H(w)]^T$$

kde $E(t)$, $F(t)$, $G(t)$, $H(t)$ sú kubické Bernsteinove polynómy definované nasledovne:

$$E(t) = (1 - t)^3$$

$$F(t) = 3 \cdot t \cdot (1 - t)^2$$

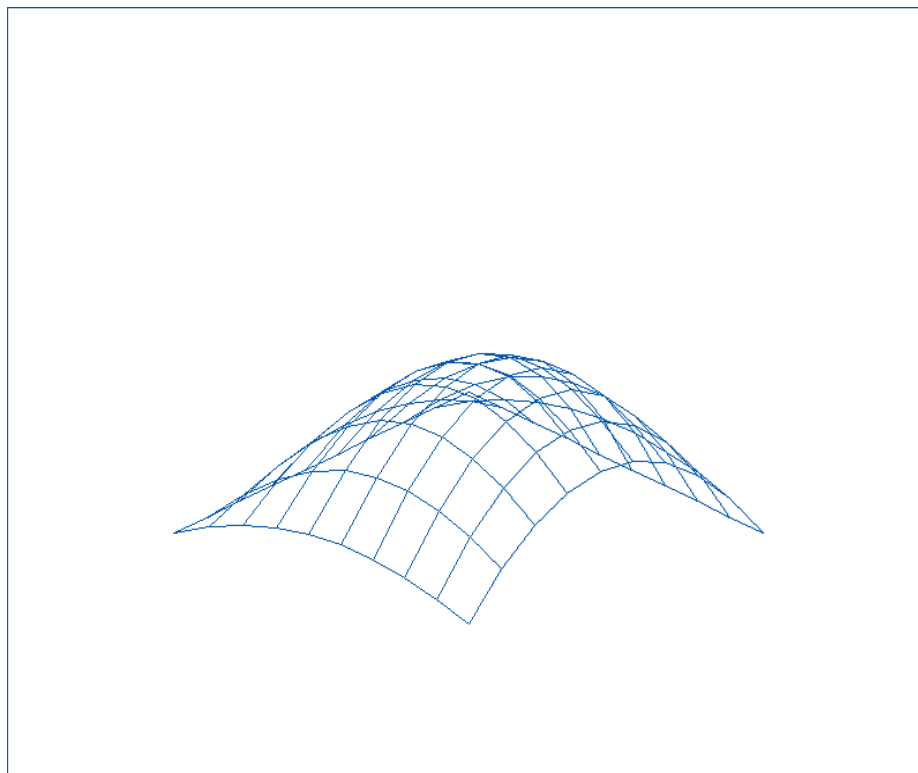
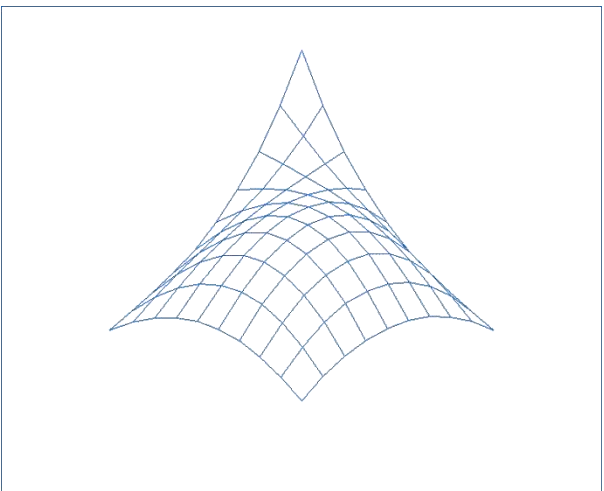
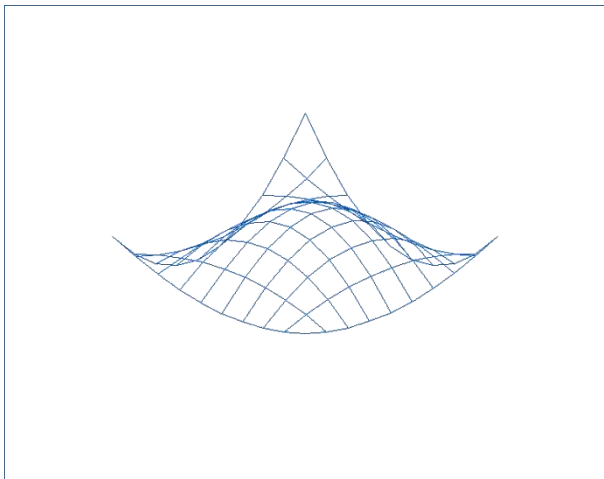
$$G(t) = 3 \cdot t^2 (1 - t)$$

$$H(t) = t^3$$

$\mathbf{B}$ je matica vrcholov
riadiacej siete:

$$\mathbf{B} = \begin{bmatrix} B_{00xyz} & B_{01xyz} & B_{02xyz} & B_{03xyz} \\ B_{10xyz} & B_{11xyz} & B_{12xyz} & B_{13xyz} \\ B_{20xyz} & B_{21xyz} & B_{22xyz} & B_{23xyz} \\ B_{30xyz} & B_{31xyz} & B_{32xyz} & B_{33xyz} \end{bmatrix}$$

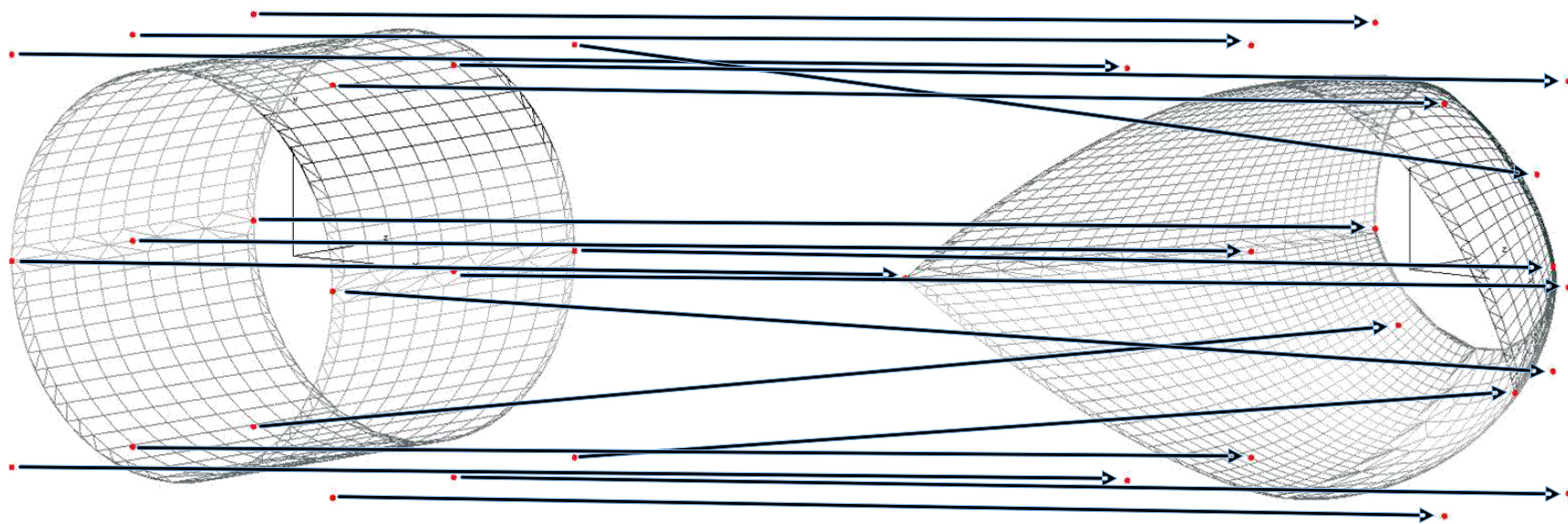
BÉZIEROVA BIKUBICKÁ PLOCHA - PRÍKLADY



MORPHING

3D, MORFOVANÉ POLOHOU RIADIACICH BODOV

- Lineárne prechody
- Nelineárne prechody

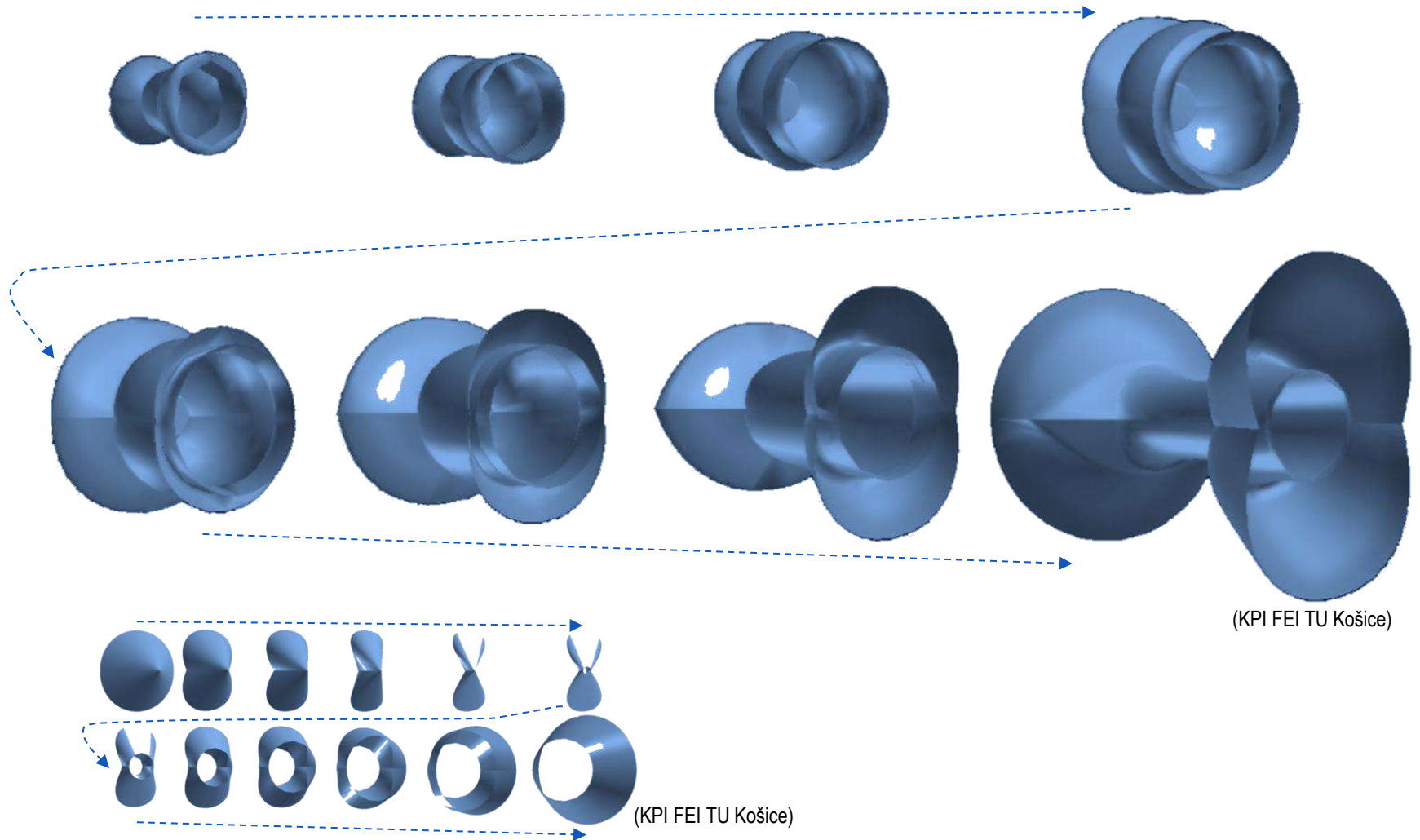


Lineárny prechod riadiacich bodov



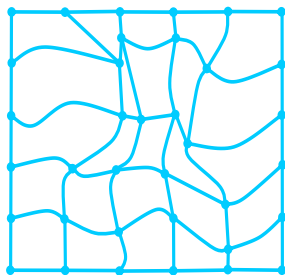
MORPHING

3D, MORFOVANÉ POLOHOU RIADIACICH BODOV

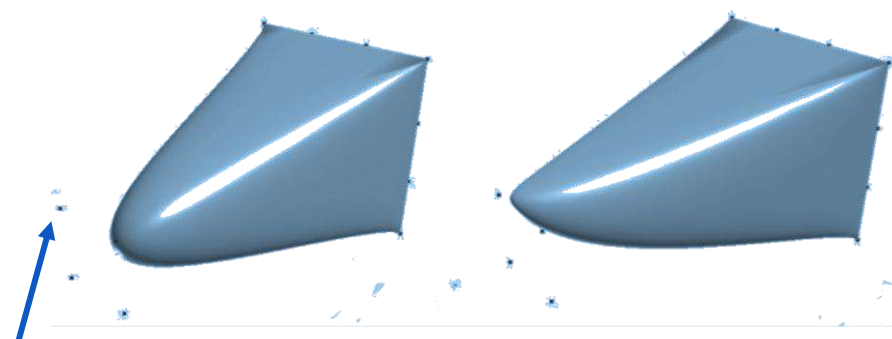


WARPING

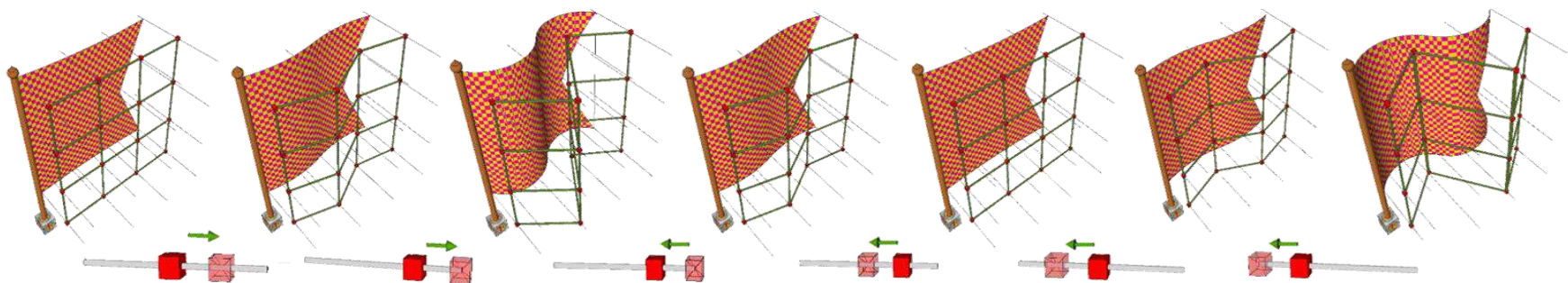
- 2D - Deformačná sieť - nelineárna
- 3D – Deformovanie pomocou zmeny polohy radiacích bodov, zmenou váh alebo uzlového vektora



Nelineárna
deformačná
sieť



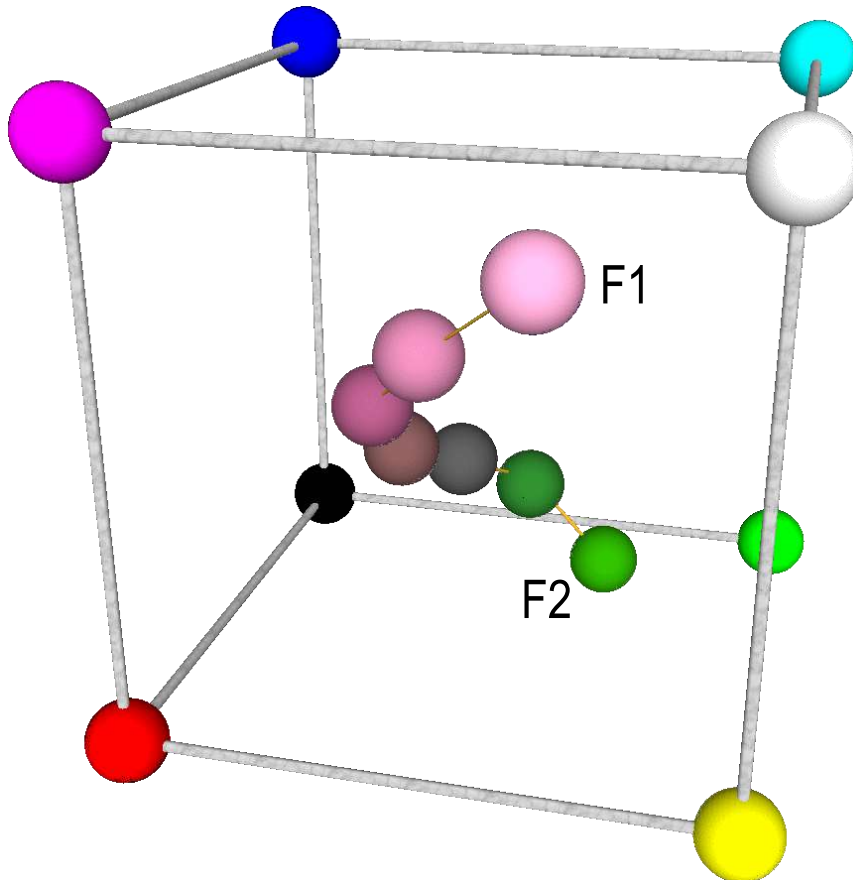
Deformovanie zmenou váhy radiaceho bodu



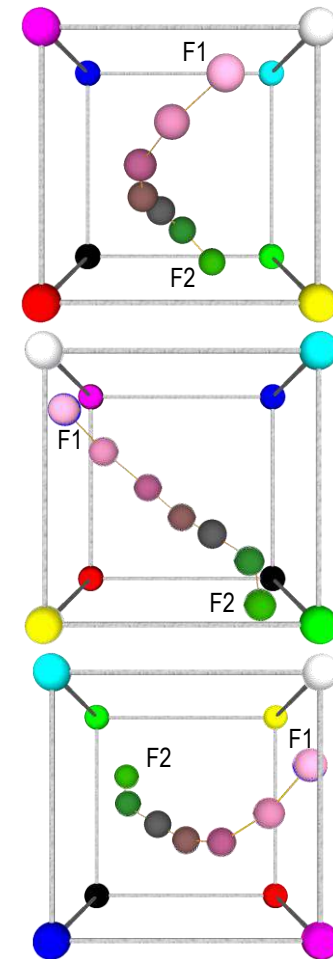
Ukážka riadenia povievania zástavy pomocou zmeny polohy radiacích bodov bikubickej plochy

PRÍKLADY POUŽITIA KRIVKY VO FAREBNOM PRIESTORE

NELINEÁRNE ALFA-MIEŠANIE



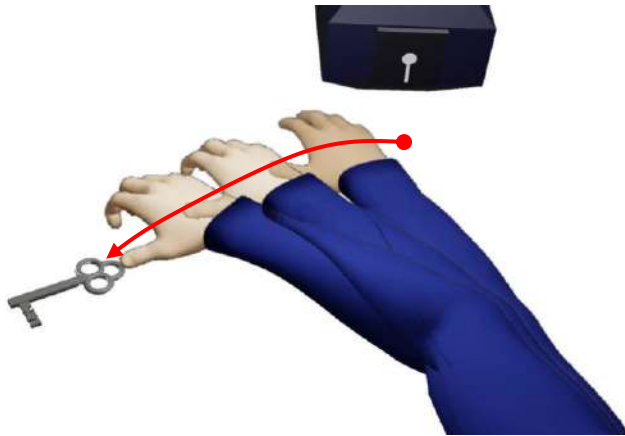
nelineárny prechod $F(\alpha)1-F(\alpha)2$ pri alfa
miešaní, RGB priestor



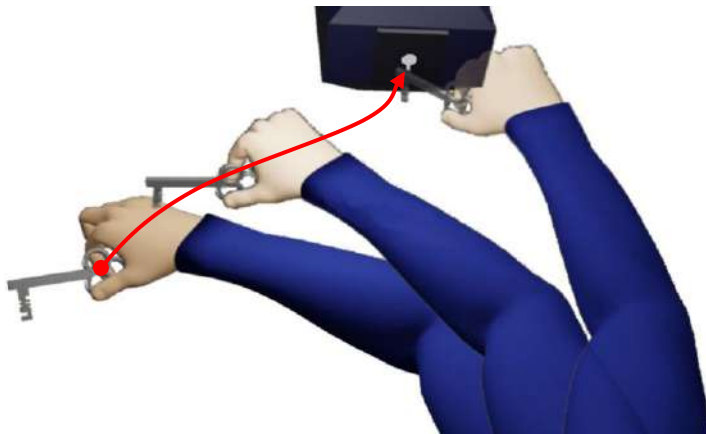
(zdroj KPI FEI TU Košice)

PRÍKLADY TRAJEKTÓRIE POHYBU RUKY

SPRACOVANIE TRAJEKTÓRIÍ MODELOVANIA POHYBOV OSÔB



(LIRKIS KPI FEI TU Košice)

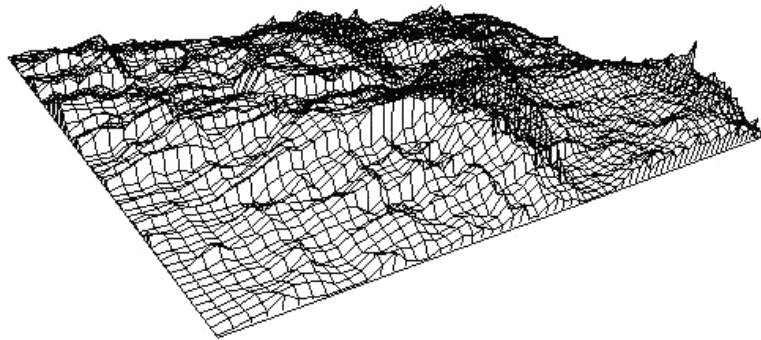
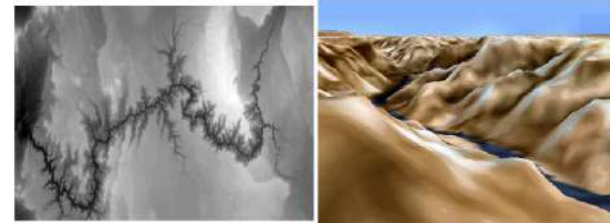
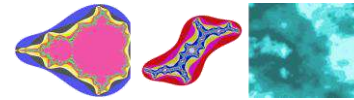


Príklad pohybu ruky po trajektórii definovanej pomocou krivky
(rehabilitácia paretickej ruky, inverzná kinematika, pohľad pacienta)

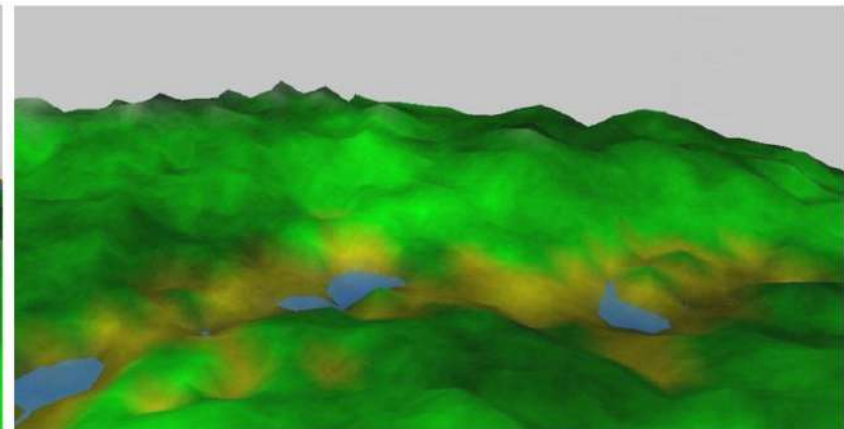
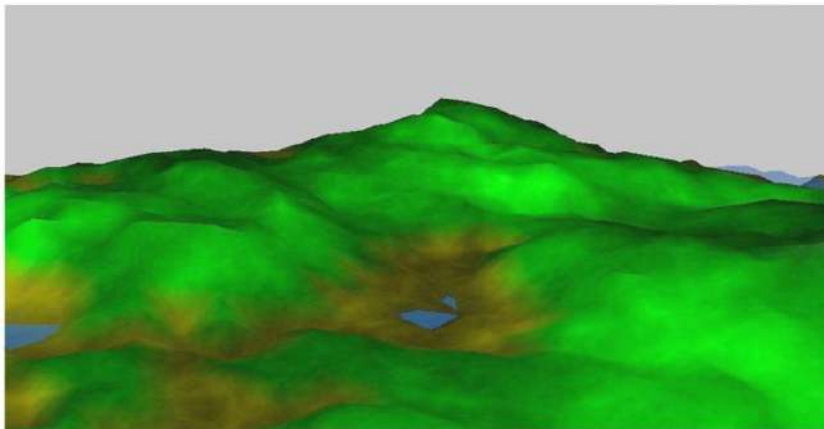
PRÍKLADY POUŽITIA POVRCHOV

Digitálny terén:

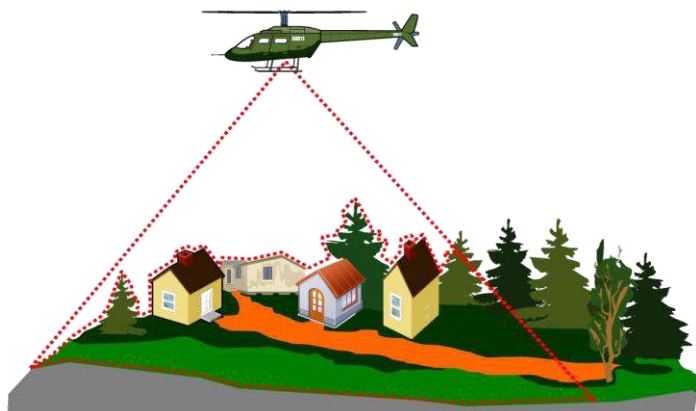
- drôtový, polygonálny,
- preložený plochou vyššej úrovne



(KPI FEI TU Košice)



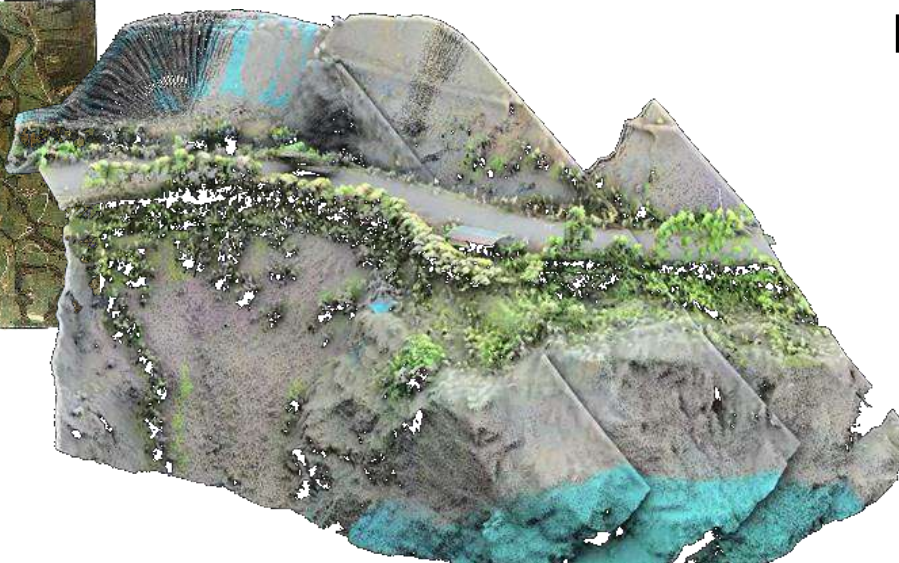
PRÍKLADY POUŽITIA PLÔCH PRI SPRACOVANÍ VÝSLEDKOV 3D SKENOVANIA TERÉNU



Lidar



(zdroj <https://www.microgeo.it/>)



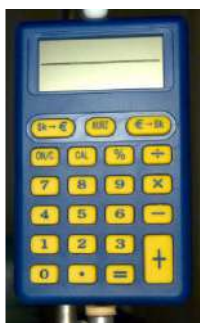
Laserový 3D skener



(LIRKIS KPI FEI TU Košice)

PRÍKLADY POUŽITIA PLÔCH

PRI SPRACOVANÍ VÝSLEDKOV 3D SKENOVANIA A PRI 3D TLAČI



(a)



(b)



(c)



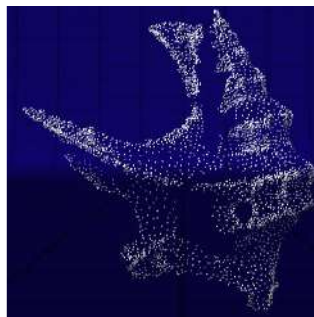
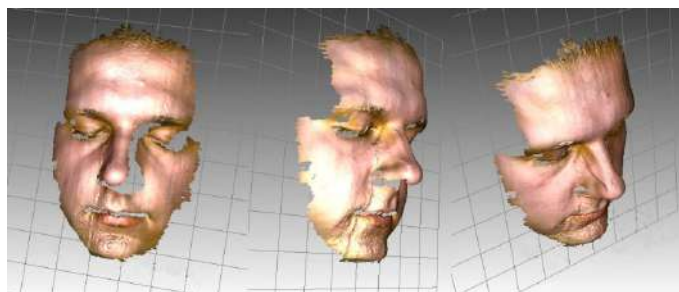
(d)



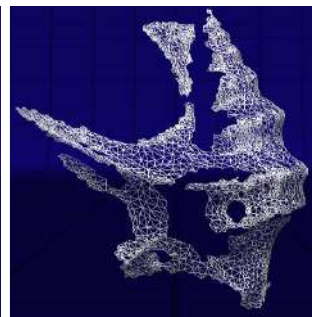
(e)



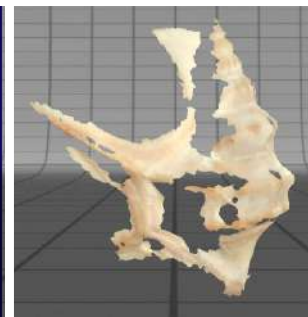
(LIRKIS KPI FEI TU Košice)



(a)



(b)



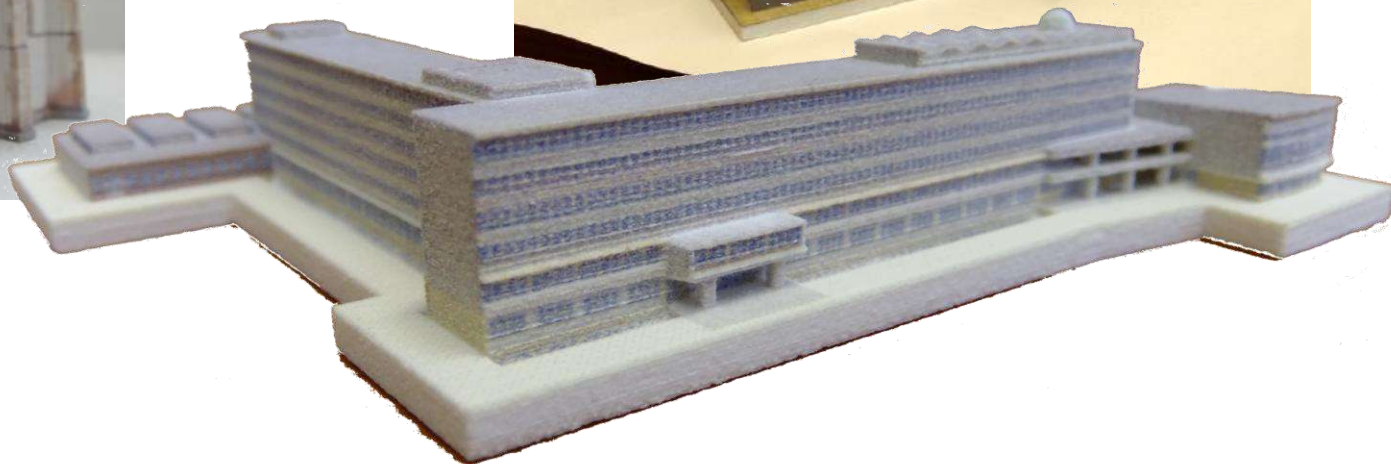
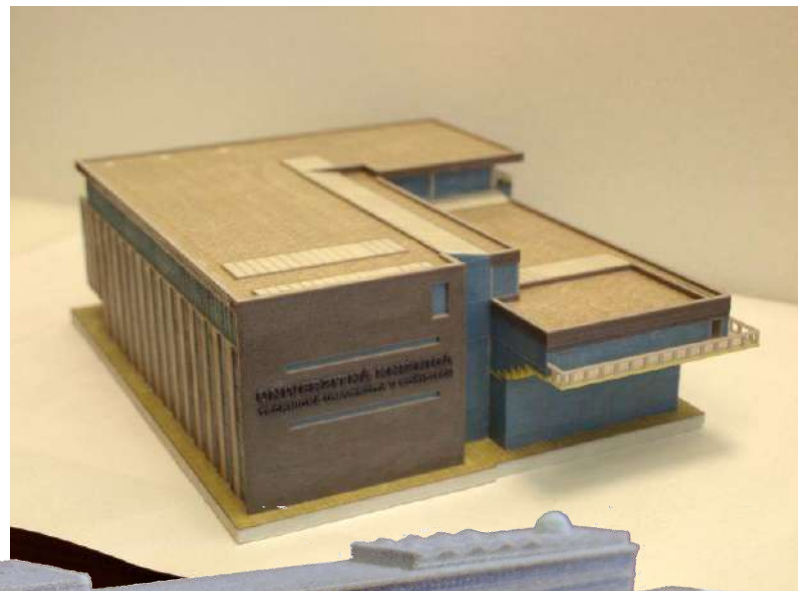
(c)

PRÍKLADY POUŽITIA PLÔCH

PRI SPRACOVANÍ VÝSLEDKOV 3D SKENOVANIA A PRI 3D TLAČI



(LIRKIS KPI FEI TU Košice)



OSTATNÉ VYUŽITIE KRIVIEK A PLÔCH

- pre použitie v časových osiach animácií
- ako vstupné parametre pri simulácii prírodných tvarov a javov
- ako prechodové krivky pri práci s farebným priestorom (napr. pri alfa-miešaní)
- ako základ pre evolučné a genetické algoritmy
- pri spracovaní trajektórií v prípade rôznych fyzikálnych simulácií vrátane inverznej kinematiky a modelovania pohybov osôb a živočíchov (vrátane mimických)



Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

RIEŠENIE VIDITEĽNOSTI V POČÍTAČOVEJ GRAFIKE

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 07

© 2025

VRSTVY VIZUALIZAČNÉHO PROCESU

1. Definovanie/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



RIEŠENIE VIDITEĽNOSTI (VISIBILITY)

Spočíva v odstránení (resp. odlíšení) tých častí trojrozmerných objektov, ktoré pri danom premietaní do 2D nie sú z miesta pozorovateľa viditeľné. Tým sú tieto časti akože zakryté a dostávame jednoznačné priemety želaných telies.



RIEŠENIE VIDITEĽNOSTI - KATEGORIZÁCIA

- podľa priestoru, kde je viditeľnosť riešená:
 - riešenie v 3D
 - riešenie v 2D priemetne
- podľa reprezentácie objektov, ktorých viditeľnosť riešime:
 - *objektovo orientované algoritmy* - kde sa rieši, ktorá časť príslušného objektu je viditeľná.
 - *obrazovo orientované algoritmy* - kde sa rieši spätne pre každý obrazový bod, ktorý objekt je v ňom vidieť.

RIEŠENIE VIDITEĽNOSTI - KATEGORIZÁCIA

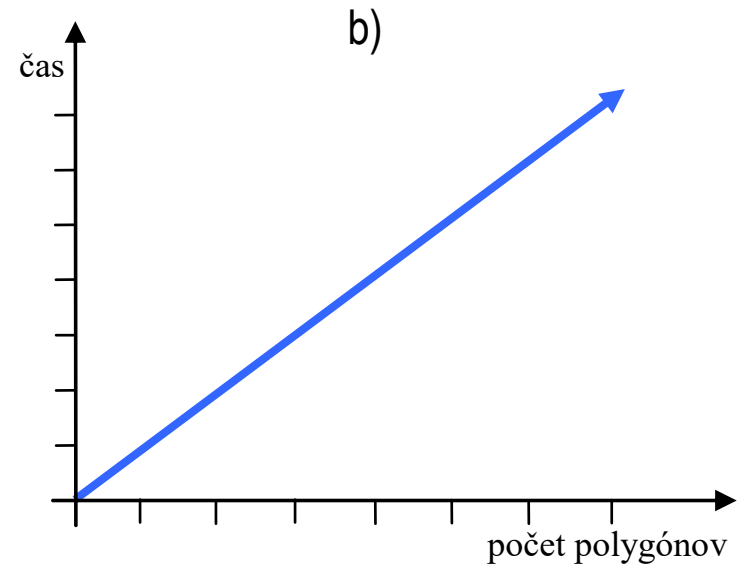
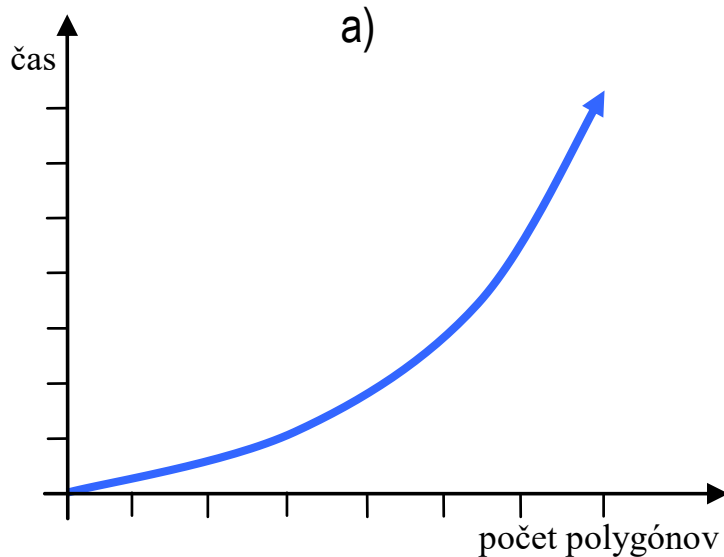
- podľa toho či sa uvažuje aj osvetlenie telesa:
 - *riešenie bez osvetlenia* - vyhodnotenie farieb je aplikované lokálne na každý objekt.
 - *riešenie s osvetlením* - medzi tieto radíme v podstate aj metódu sledovania lúča (raytracing) alebo vyžarovaciu metódu (radiosity), ktoré sa častokrát označujú aj ako globálne metódy riešenia viditeľnosti s globálnou aplikáciou farieb v rámci scény t.j. napr. aj odrazy.

RIEŠENIE VIDITEĽNOSTI - KATEGORIZÁCIA

- podľa vplyvu možnej chyby pri vykonávaní (napr. použitím celočíselnej aritmetiky):
 - *s lokálnym vplyvom chyby na výsledok*
 - *s globálnym vplyvom chyby na výsledok*
- podľa času potrebného na riešenie viditeľnosti:
 - *riešenie mimo reálneho času*
 - *riešenie v reálnom čase*

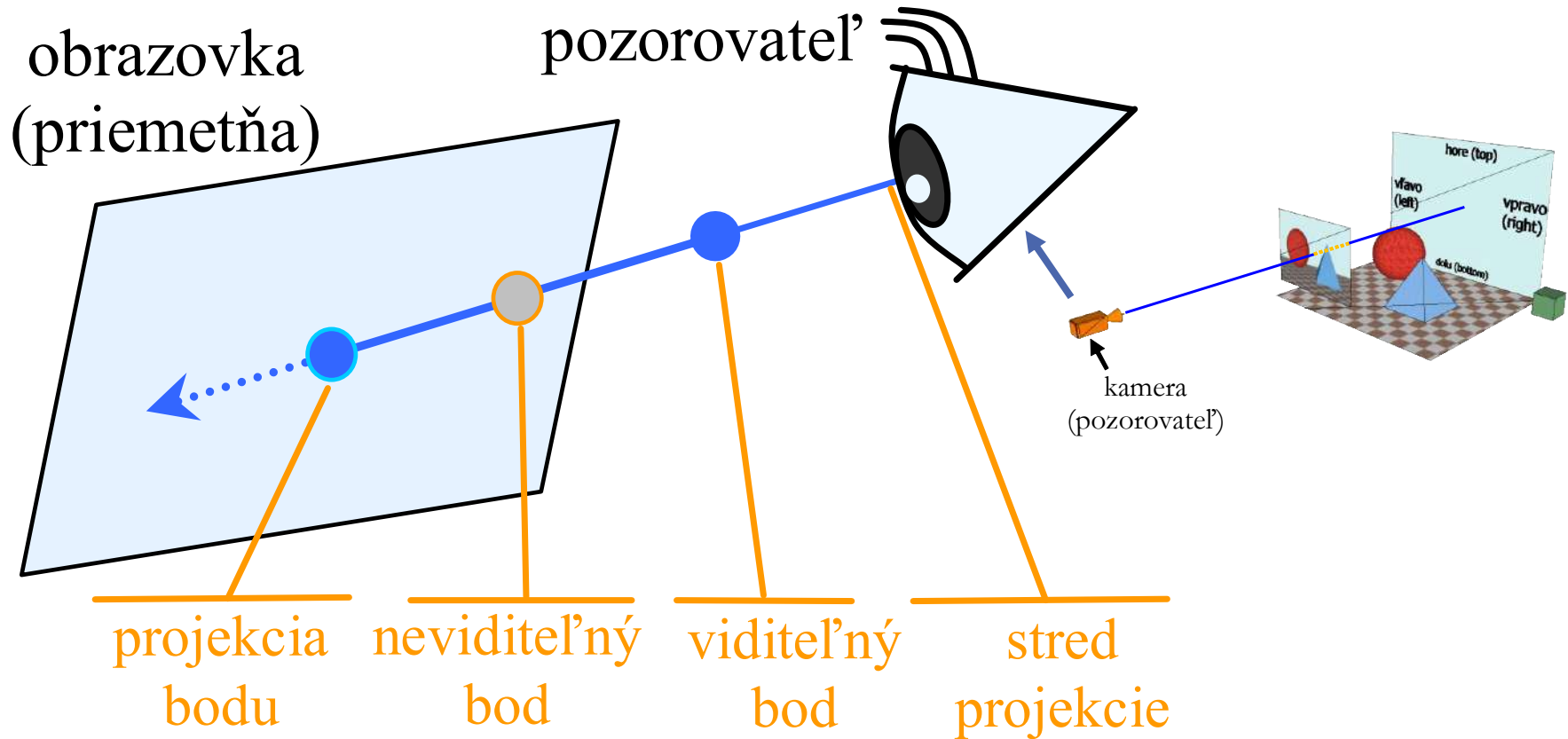
ČASOVÁ ZÁVISLOSŤ RIEŠENIA VIDITEĽNOSTI

$$T = T_{V_{\text{ýber}}} + T_{K_{\text{reslenie}}}$$



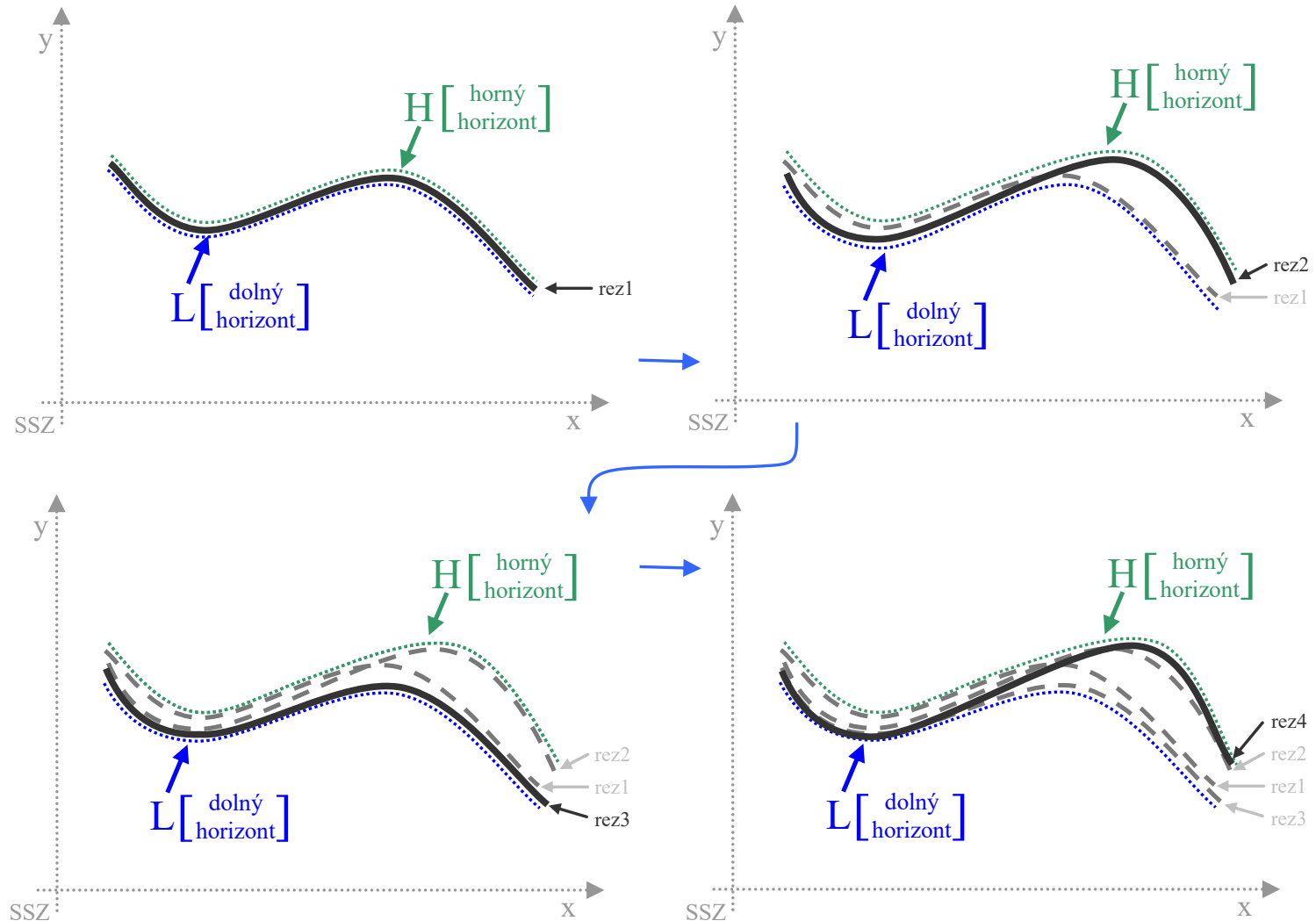
T_v pre a) objektové a b) rastrové algoritmy

VIDITEĽNOSŤ BODU



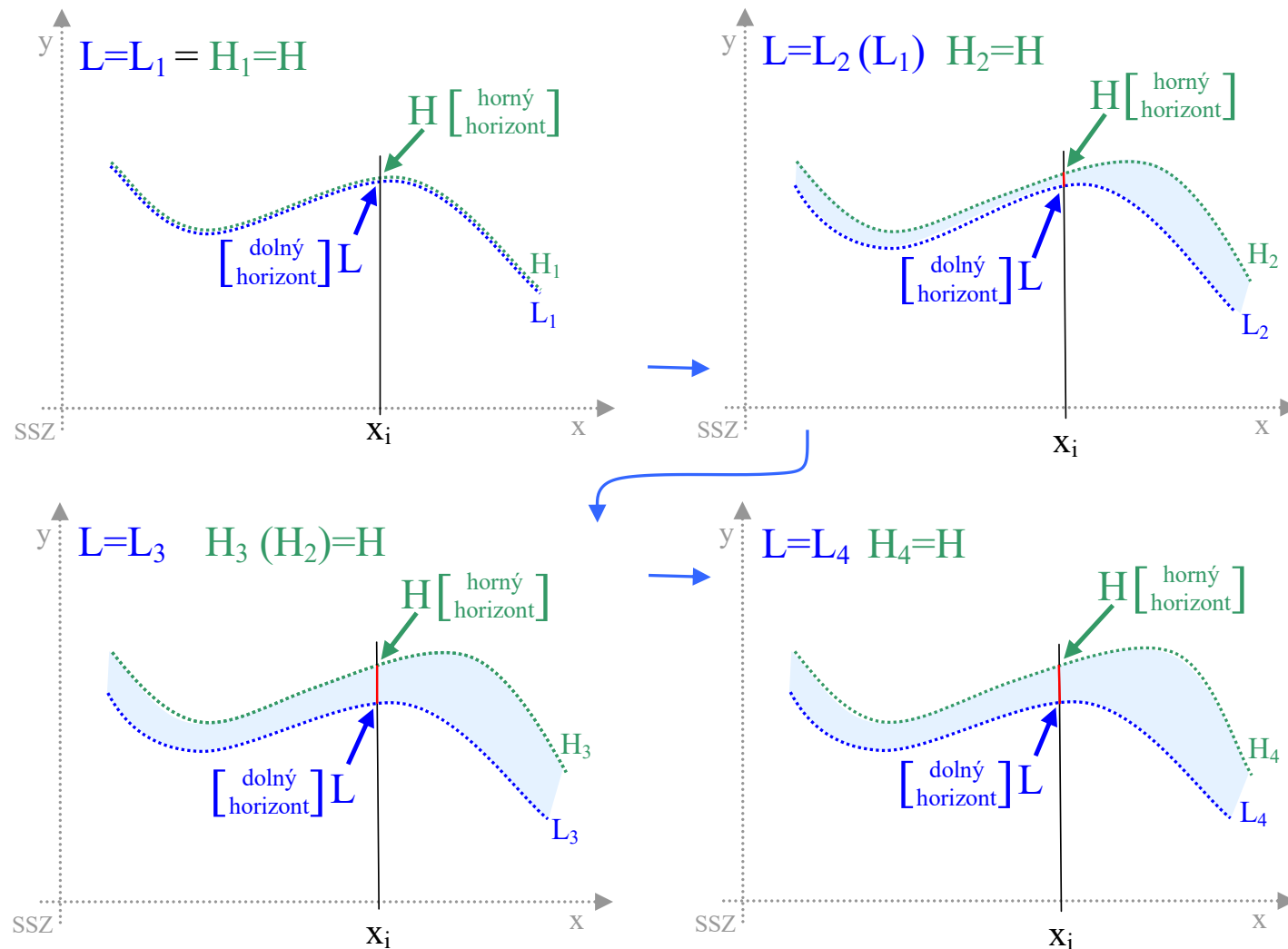
ALGORITMUS PLÁVAJÚCEHO HORIZONTU

NA ÚROVNI REZOV/KRIVIEK



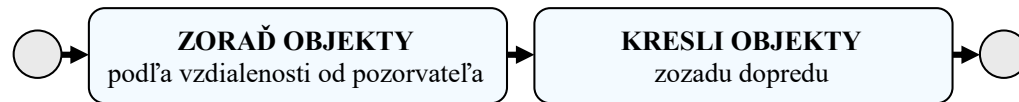
ALGORITMUS PLÁVAJÚCEHO HORIZONTU

NA ÚROVNI INTERVALU HODNÔT V X_i

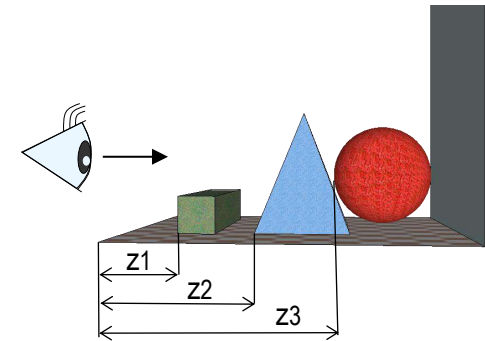


MALIAROV ALGORITMUS

PAINTER'S ALGORITHM



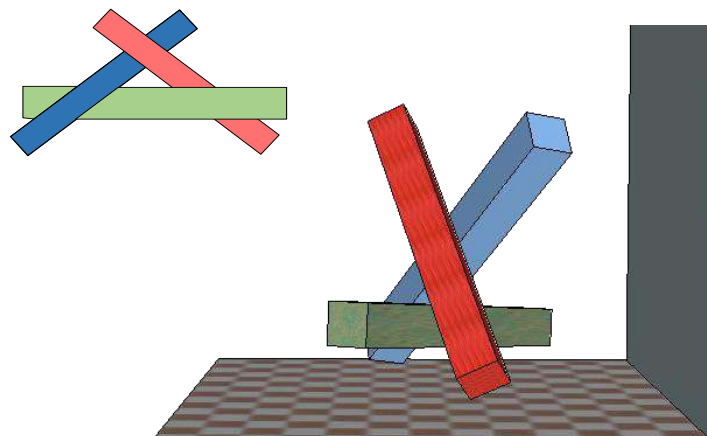
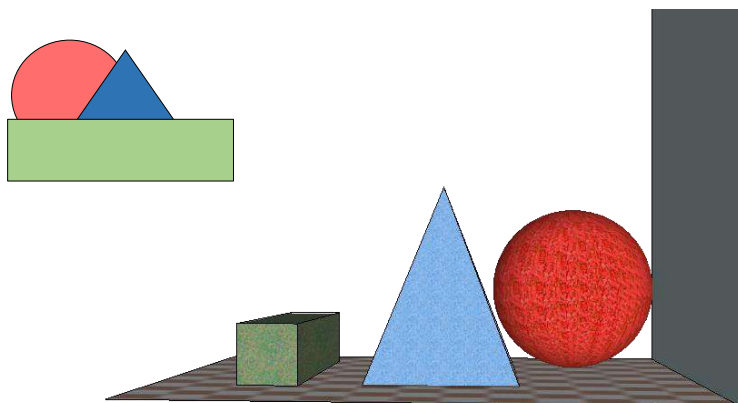
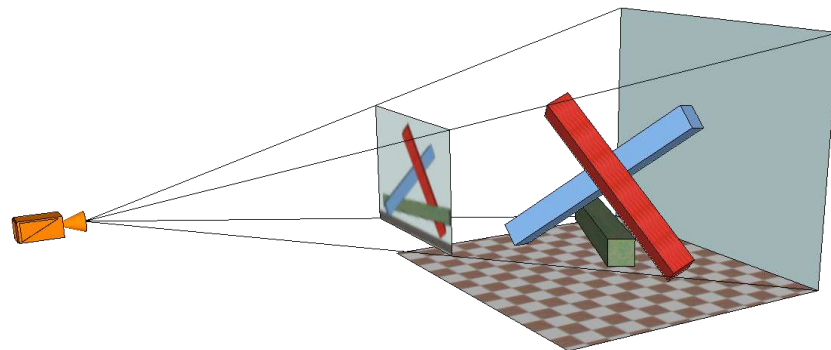
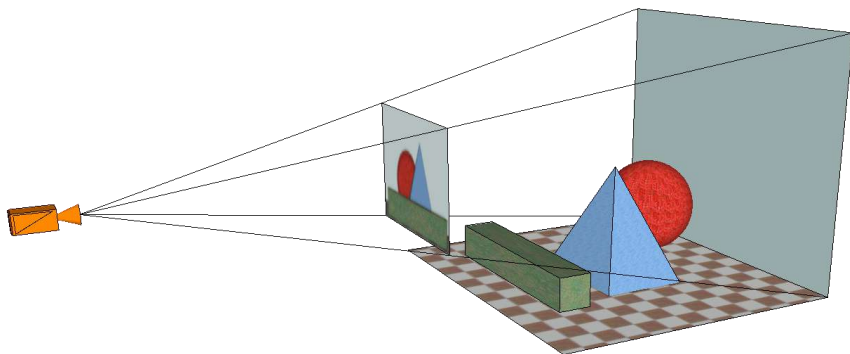
(Z-Sort, $O(n \cdot \log^2 n)$ v najlepšom prípade)



Podmienky:

- Ak $z1_{max} < z2_{min}$, potom je polygón $P1$ úplne za polygónom $P2$.
- Ak sa neprekrývajú priemety polygónov v rovine priemetne (t.j. v SSZ), je poradie ich vykresľovania nezávislé a polygón $P1$ testu vyhovuje.
- Ak sú všetky vrcholy polygónu $P1$ pod rovinou určenou polygónom $P2$, potom $P1$ leží za polygónom $P2$.
- Ak sú všetky vrcholy polygónu $P2$ nad rovinou určenou polygónom $P1$, potom $P1$ leží za polygónom $P2$.

MALIAROV ALGORITMUS



Modelové scény pre maliarov algoritmus bez a s cyklickým prekryvom

WARNOCKOV ALGORITMUS

Warnockov algoritmus umožňuje zobrazit' viditeľné časti telesa dvoma spôsobmi:

- hrany (drôtové zobrazenie)
- steny (plošné zobrazenie)

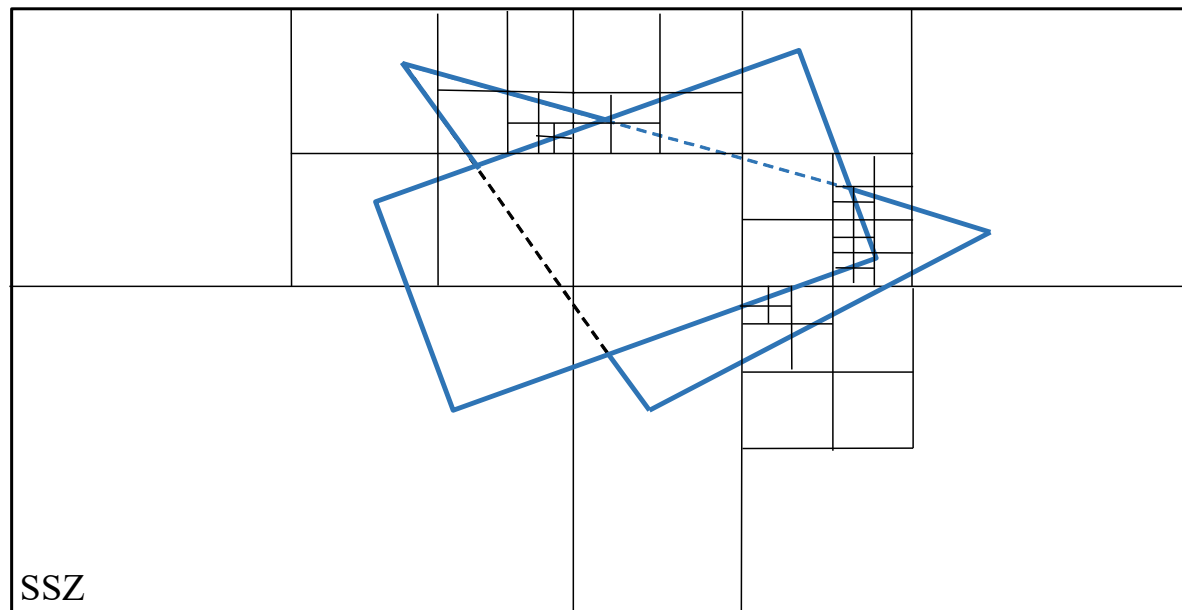


WARNOCKOV ALGORITMUS

1. Uloženie základného zobrazovacieho priestoru, teda jednej z tých 4.častí do zásobníka.
2. Výber zo zásobníka a sprístupnenie informácií, ktoré definujú veľkosť okna.
3. Ohodnotenie a rozhodnutie, vzhľadom na jednotlivé možnosti, ktoré môžu nastať pri kreslení telies po delení v príslušnom obdĺžniku (okne). Možnosti:
 - a) V okne sa nachádza mnohouholník prekrývajúci daný obdĺžnik.
 - b) Mnohouholník je mimo obdĺžnika.
 - c) Ostatné prípady.
4. Keď sa jedná o možnosti a) alebo b), potom vieme jednoznačne rozhodnúť o viditeľnosti (výpočet vzdialenosti a rozhodnutie je v 3D). Nastáva vykreslenie.
5. Keď nevieme rozhodnúť o viditeľnosti, teda jedná sa o možnosť c), potom sa robí ďalšie (vlastne rekurzívne) delenie príslušného okna na 4 rovnaké časti. Vypočítajú sa súradnice jednotlivých okien a každé okno sa uloží do zásobníka.

WARNOCKOV ALGORITMUS

6. Keď rozmery okna sú rovné jednej, tzn. že sa jedná o bod (pixel), dá sa jednoznačne rozhodnúť, ktorý bod je viditeľný a ten sa vykreslí.
7. Keď sa zásobník vyprázdni, ukončí sa celý algoritmus.
8. V opačnom prípade sa pokračuje v kroku 2.



Postup delenia pri Warnockovom algoritme

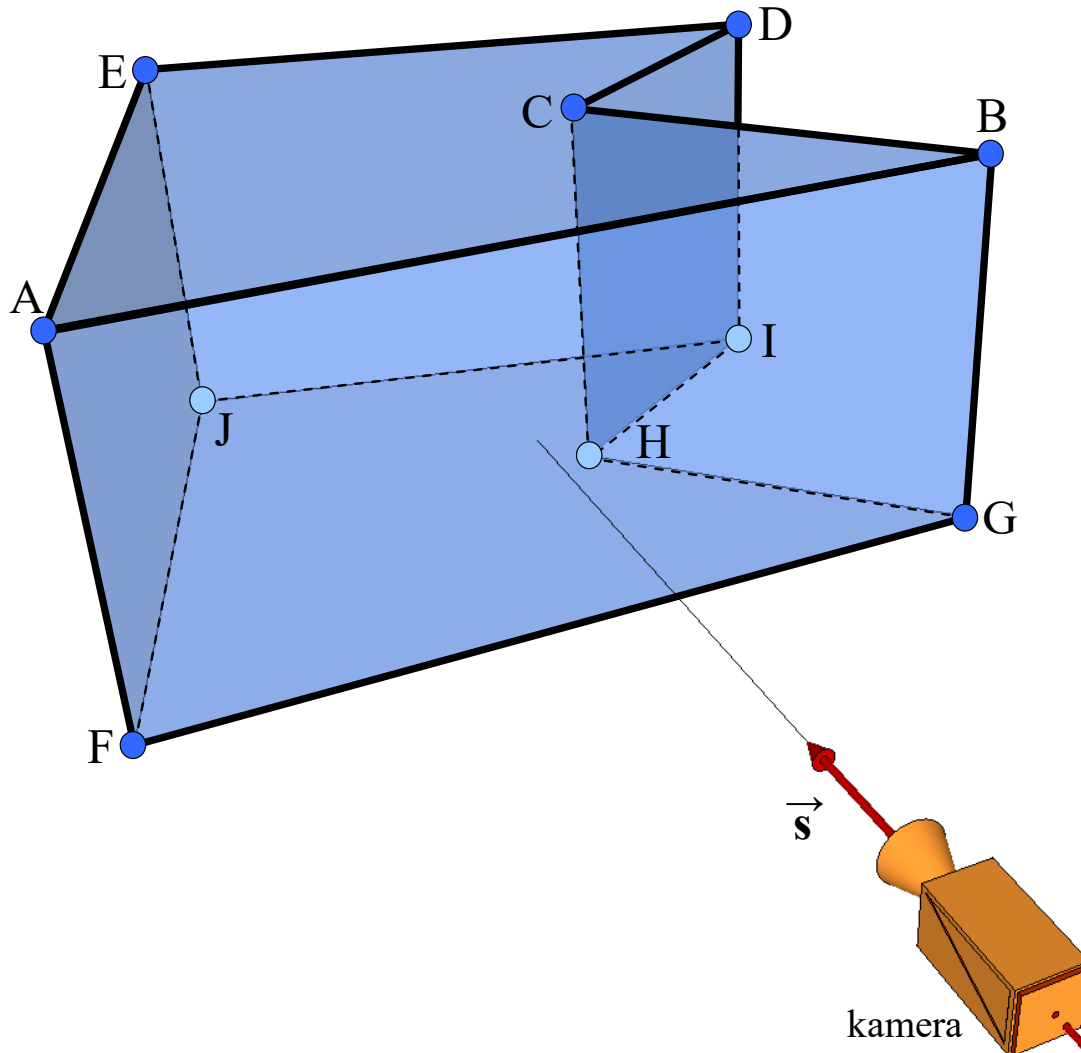
FREEMAN-LOUTRELOV ALGORITMUS

Algoritmus pracuje v 3D a je založený na rozdelení stien na neviditeľné a potencionálne viditeľné na základe uhla φ medzi vektorom pohľadu kamery $\vec{k}$ a normálou steny $\vec{n}$ a následne kreslenie zozadu dopredu len z množiny potencionálne viditeľných.

$$\varphi = \arccos \left(\frac{k_x \cdot n_x + k_y \cdot n_y + k_z \cdot n_z}{\sqrt{k_x^2 + k_y^2 + k_z^2} \cdot \sqrt{n_x^2 + n_y^2 + n_z^2}} \right)$$

Je potrebné zabezpečiť jednotnú orientáciu všetkých stien v scéne (v smere alebo proti smeru hodinových ručičiek)

FREEMAN-LOUTRELOV ALGORITMUS



Príklad typu stien:

orientácia
 $\vec{n}(U, V, X, Z)$
 normála príslušnej steny

$\vec{n}(B, C, H, G) \times \vec{s} > 0$
 zadná (neviditeľná)

$\vec{n}(C, D, I, H) \times \vec{s} < 0$
 predná
 (potencionálne viditeľná)

Podmienka:

orientácia v celej scéne
 rovnaká

ALGORITMUS PAMÄTE HLĚBKY (Z-BUFFER)

výhody:

- korektne rieši viditeľnosť
- polygóny (mnohouholníky) sa môžu navzájom pretínať
- polygóny je možné kresliť v ľubovoľnom poradí (nevyžaduje žiadne predspracovanie ani triedenie polygónov)
- vhodný pre objekty s množstvom malých polygónov

nevýhody:

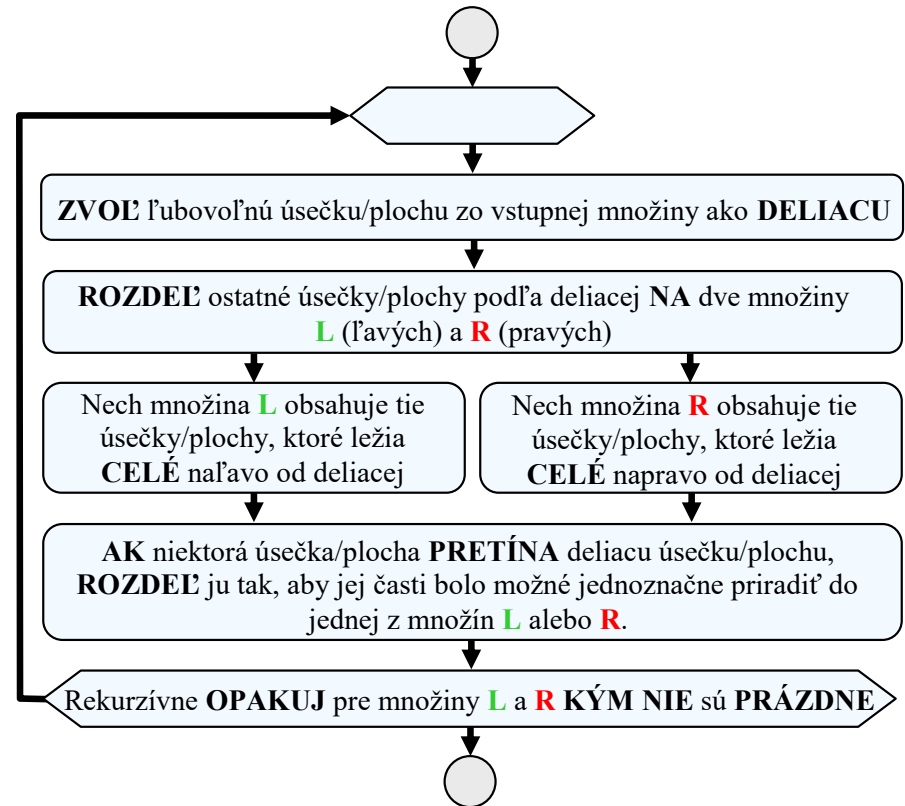
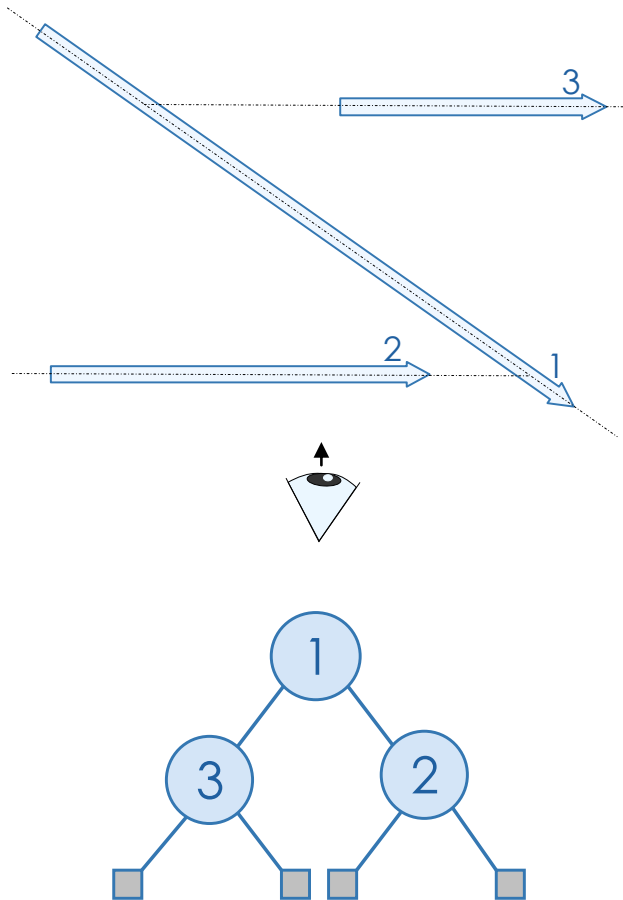
- vysoké nároky na pamäť, minimálne $(2 \cdot \max X \cdot \max Y)$ bajtov
- prekresľovanie
- je pomalý (málo efektívny) pri veľkých scénach (veľa veľkých polygónov), je ho možné však urýchliť použitím orezania na zorný ihlan.

BSP STROMY (BINARY SPACE PARTITIONING)

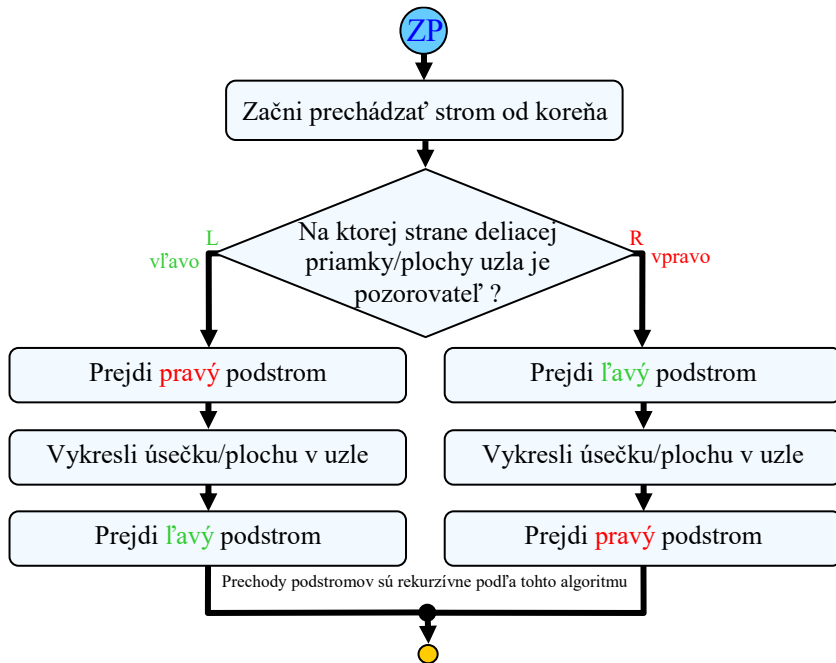
- Podľa rozloženia objektov:
 - Vyvážené
 - Nevyvážené
- Podľa rozloženia deliacich rovín:
 - Statické
 - Dynamické
 - Adaptívne
 - Stratové



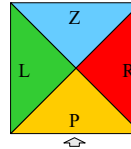
BSP STROMY



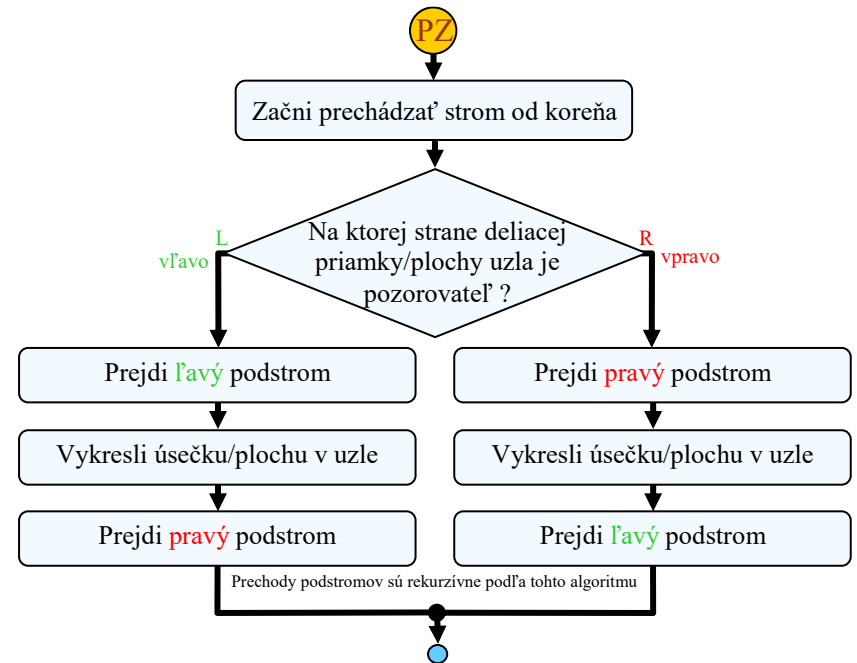
BSP STROMY - PRECHOD STROMAMI



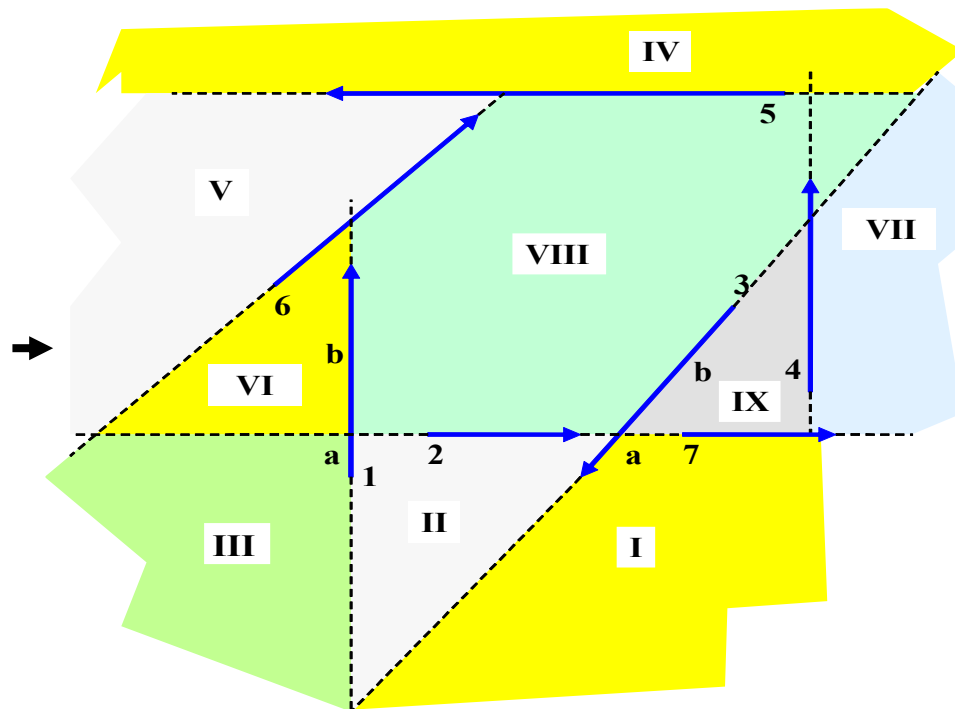
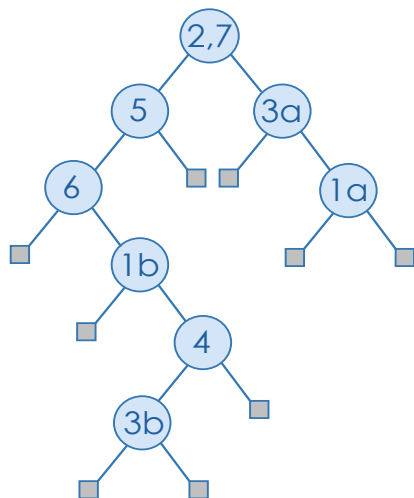
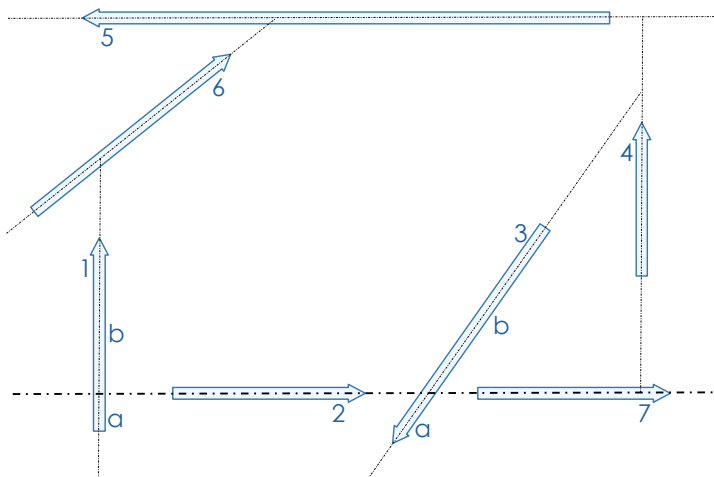
prechod **z**ozadu **dop**redu



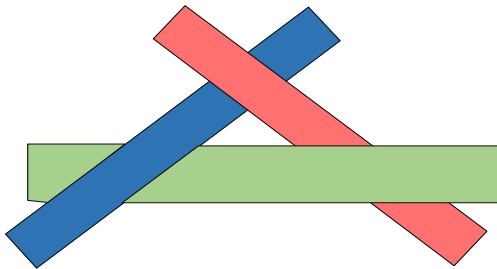
prechod **s**predu **doz**adu



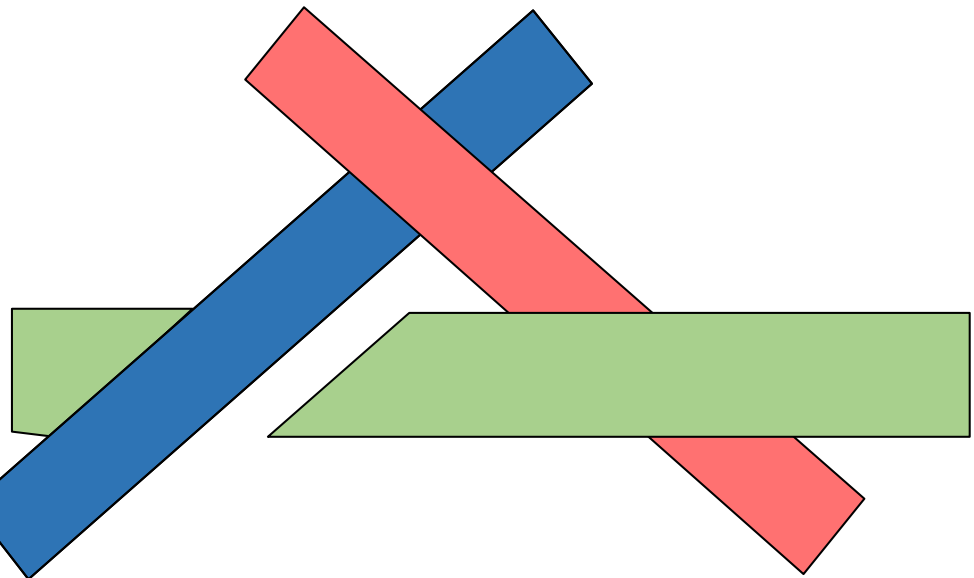
BSP STROMY



BSP STROMY – RIEŠENIE CYKLICKEJ SCÉNY

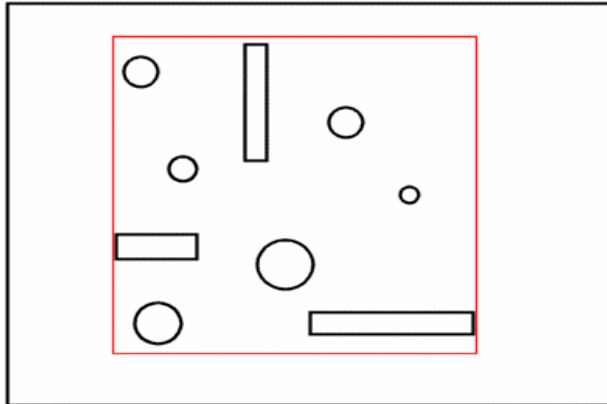


Nevýhoda: možný nárast
spracovávaných polygónov

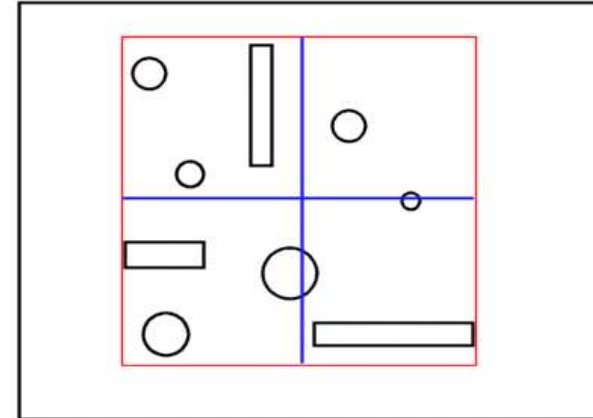


OKTANTOVÉ STROMY (OCTREES)

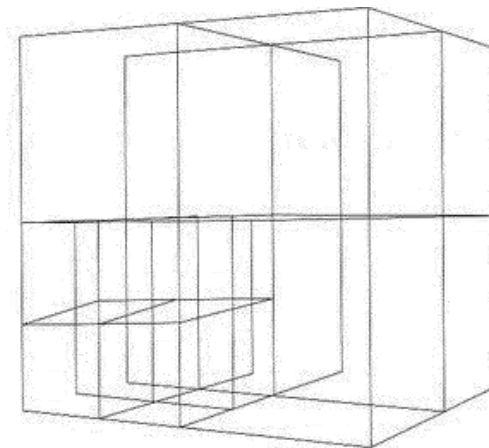
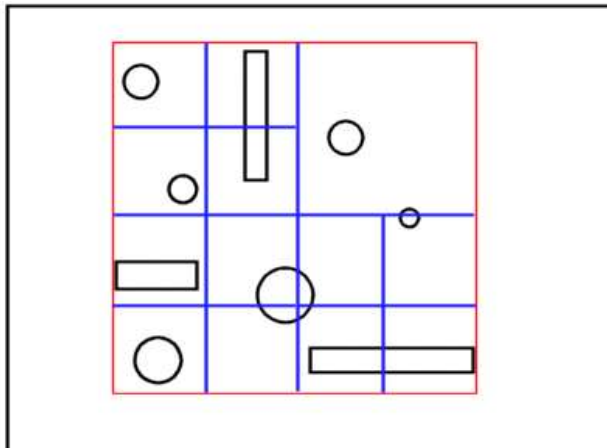
originál



1. delenie



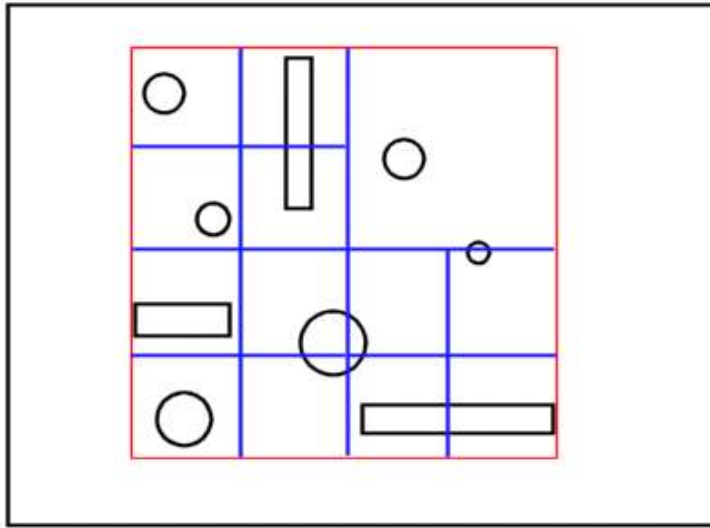
2. delenie



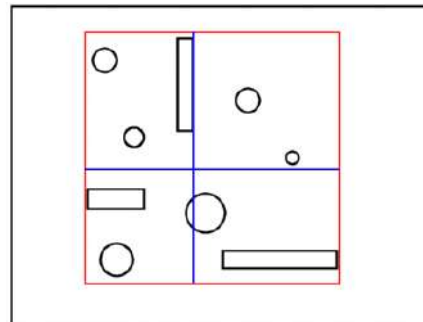
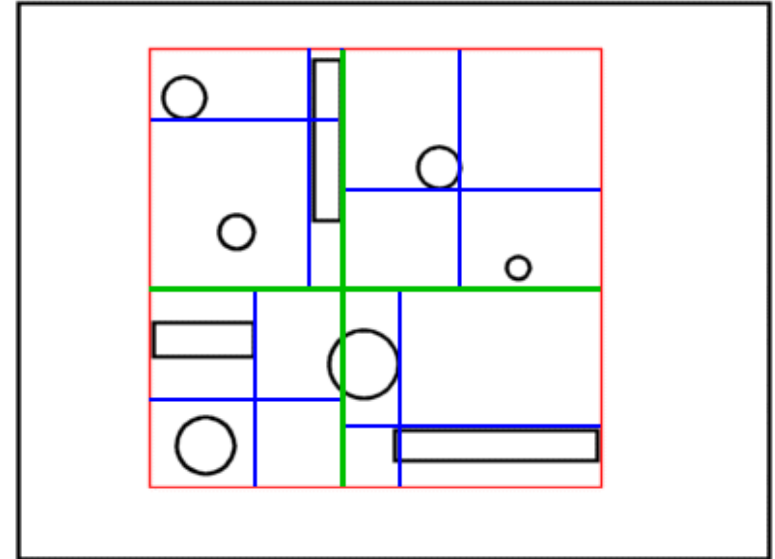
Výsledné rozdelenie

OKTANTOVÉ STROMY

Pôvodné rozdelenie

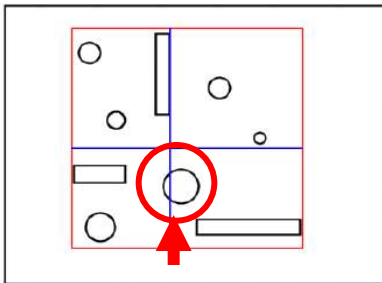
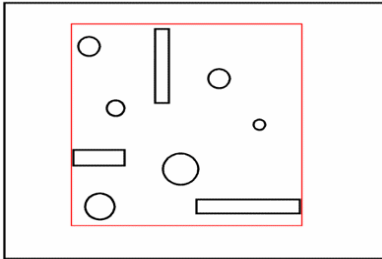


Adaptívne rozdelenie

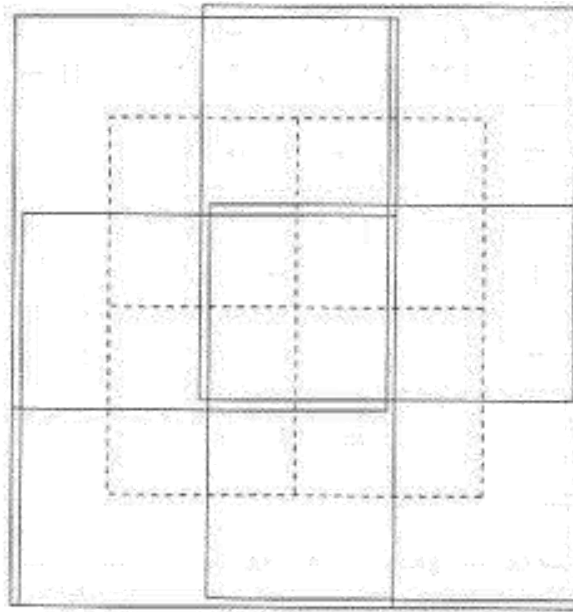


OKTANTOVÉ STROMY

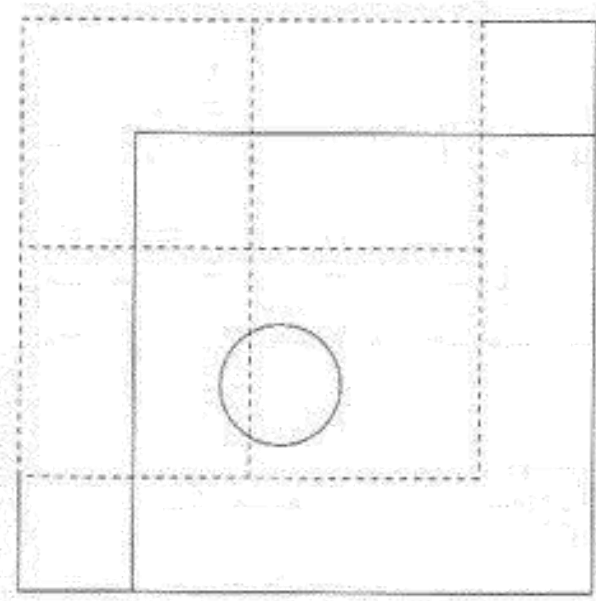
originál



Delenie pri použití
adaptívneho delenia

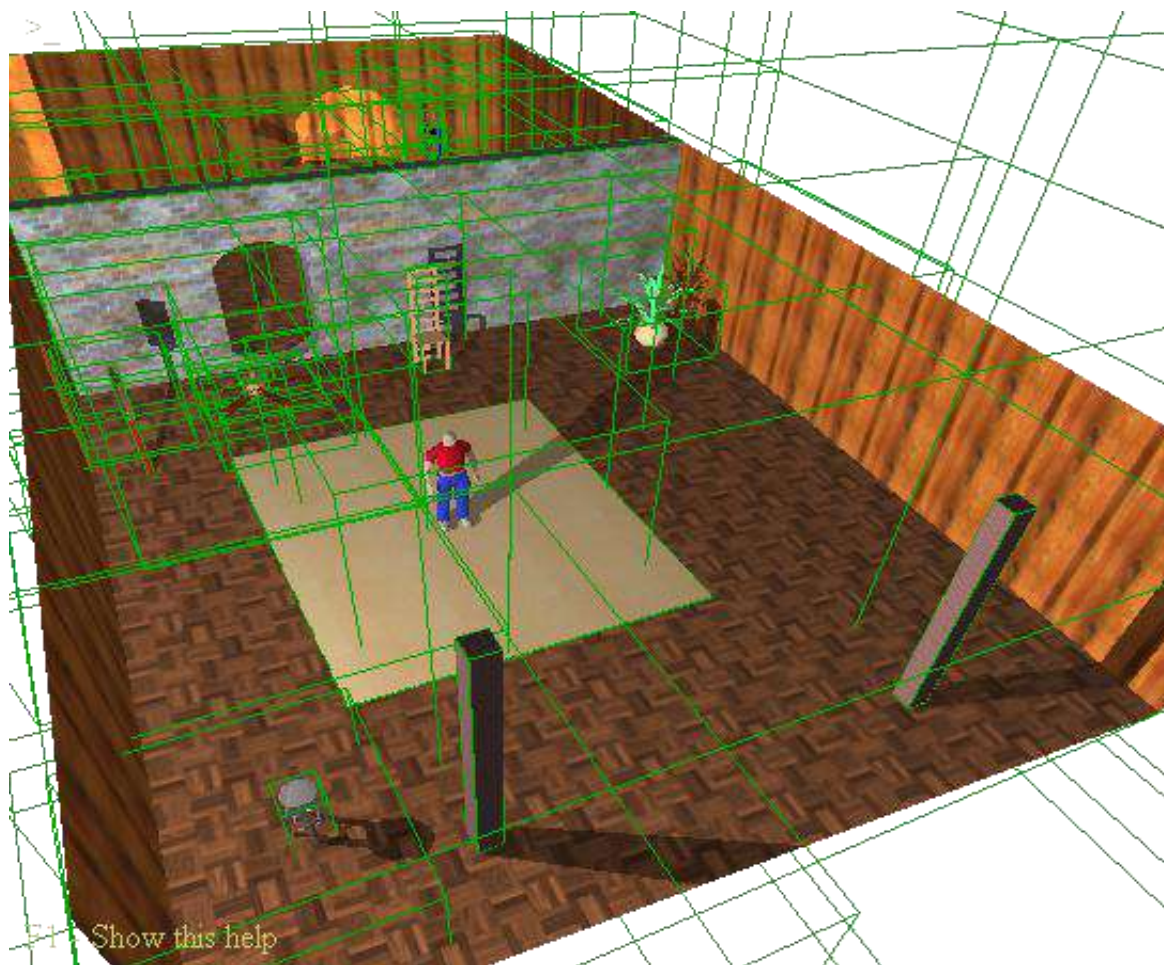


Stratové delenie



Umiestnenie
problematickeho
objektu v stratovom
oktantovom strome
(pravý dolný uzol)

OKTANTOVÉ STROMY



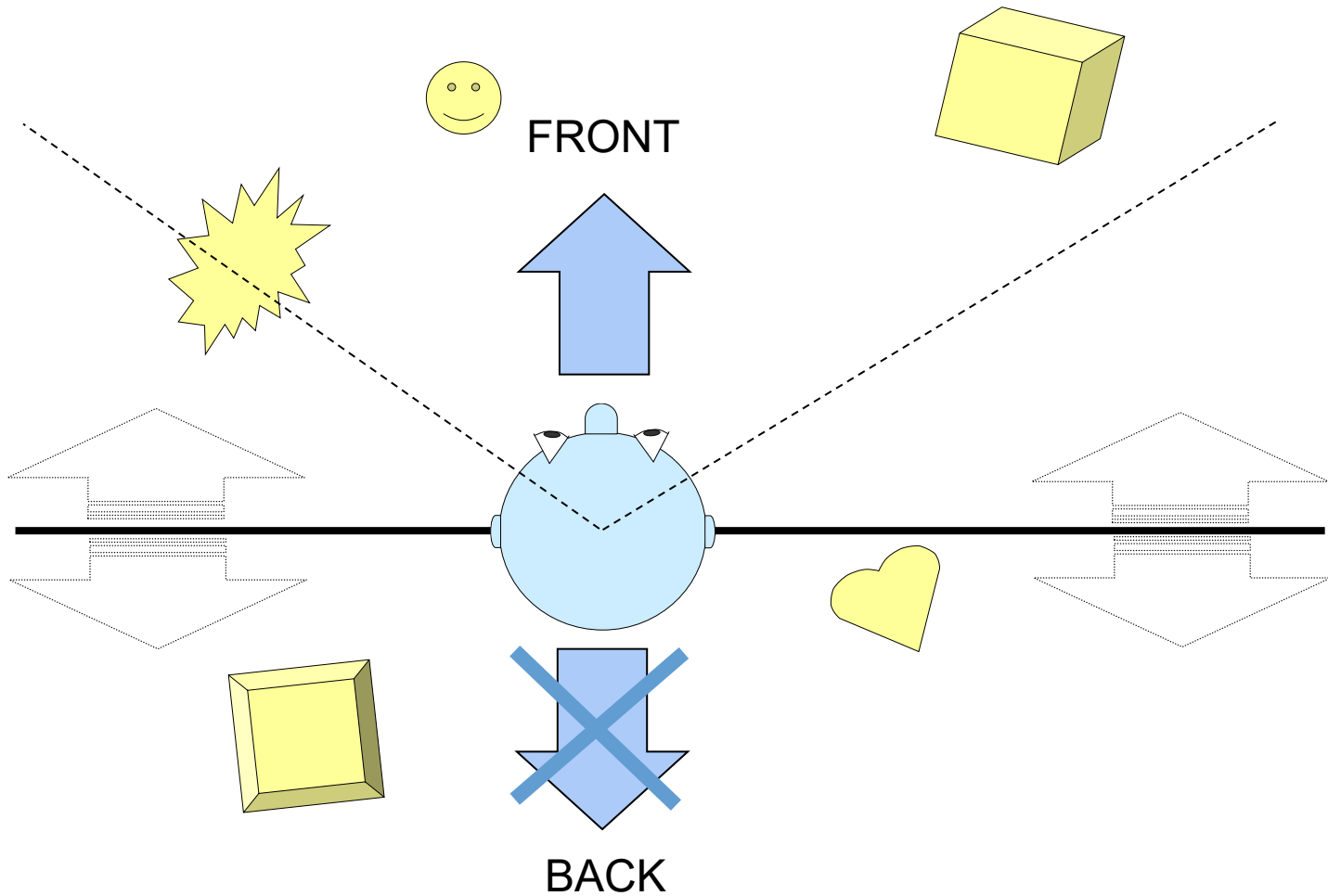
Príklad zložitejšieho
delenia priestoru
pomocou
oktantových stromov

URÝCHĽOVACIE METÓDY RIEŠENIA VIDITEĽNOSTI

- FV (Front view)/ BC (Back Cut)
- Orezávanie na zorný ihlan
- Ohraničujúce objekty
- Sektorovanie
- Potenciál viditeľnosti (PVS)
- S-buffer



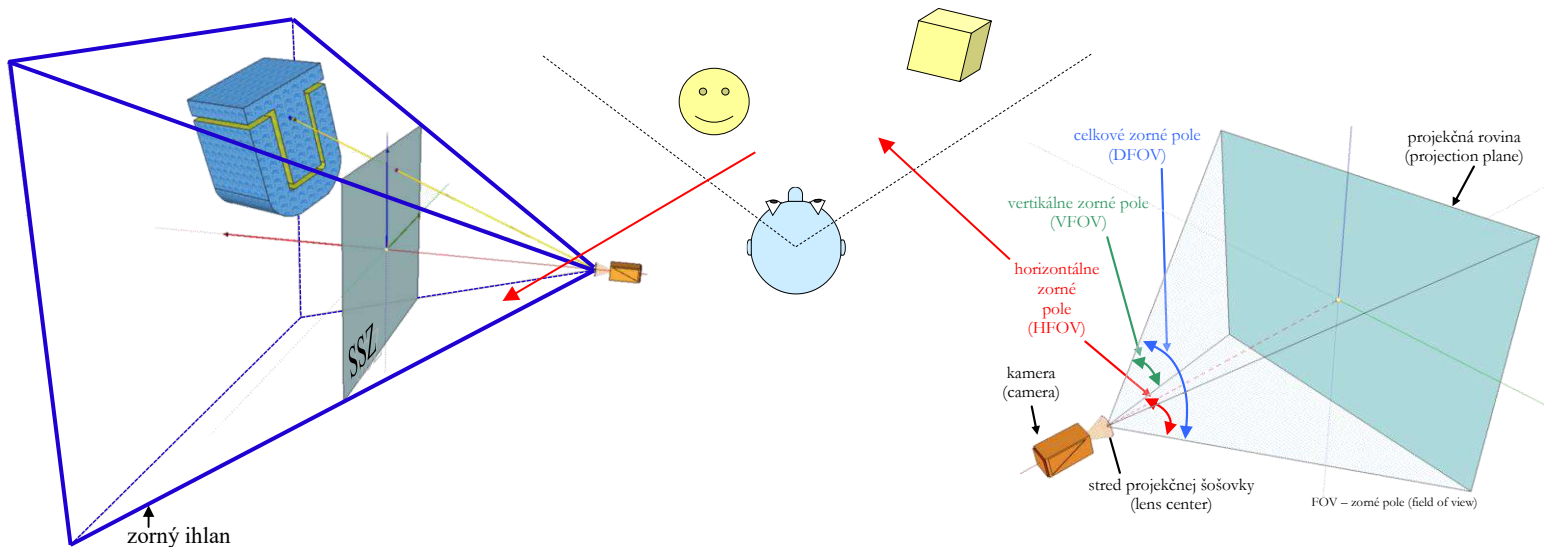
FRONT VIEW / BACK CUT



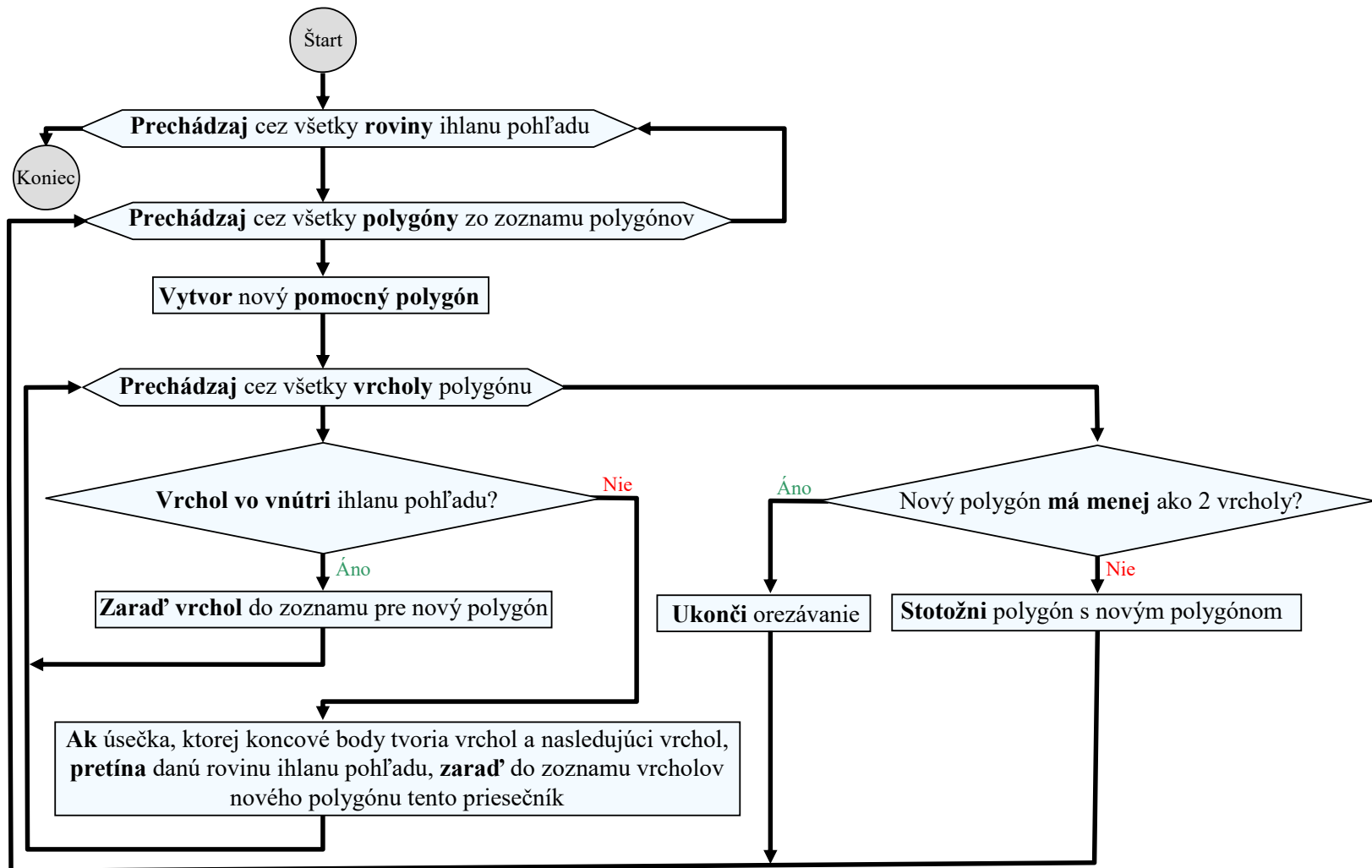
OREZÁVANIE NA ZORNÝ IHLAN

Zorný ihlan je časť priestoru, ktorá je vymedzená H/V -FOV a po projekčnej a zobrazovacej transformácii ($USS \rightarrow SSC \rightarrow SSZ$) zobrazí do obdĺžnika na priemetni, ktorý reprezentuje tú jej časť, ktorá v konečnom dôsledku bude viditeľná na zobrazovači (obrazovke) t.j. zorný priestor.

Orezávanie je možné robiť buď v SSC alebo v USS .

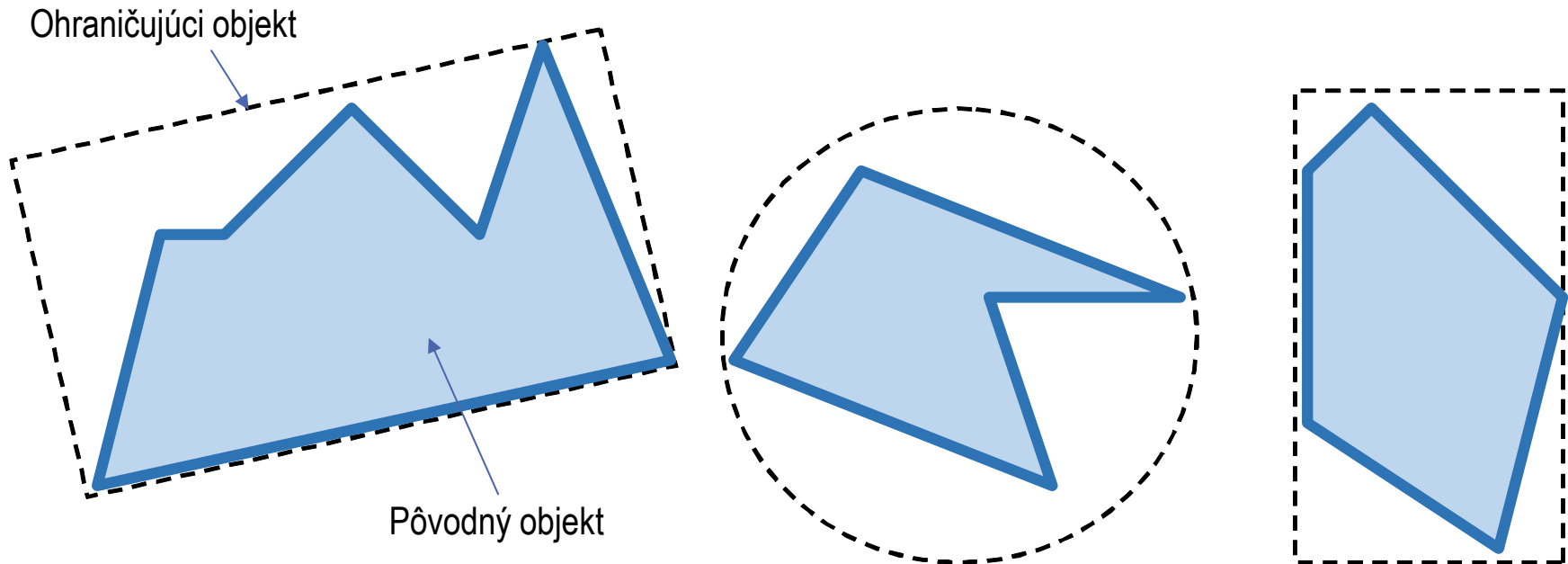


OREZÁVANIE NA ZORNÝ IHLAN - ALGORITMUS



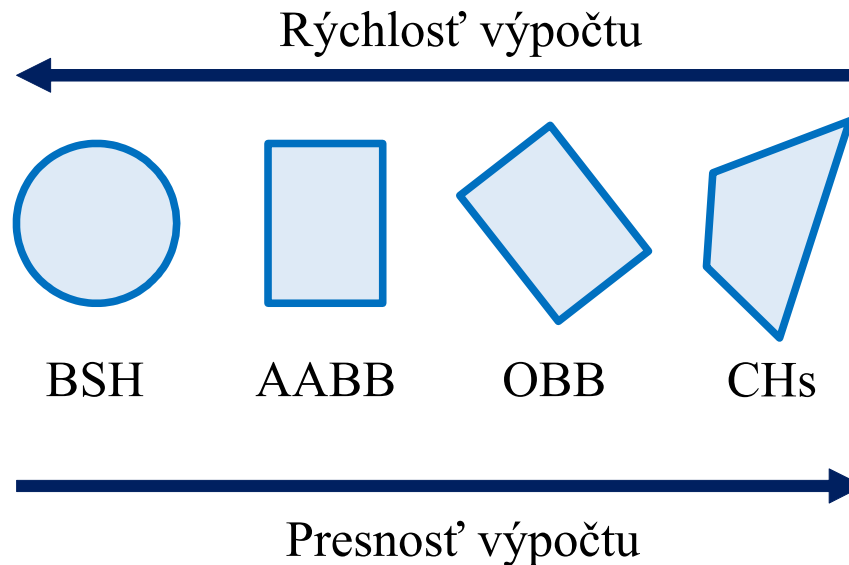
OHRANIČUJÚCE OBJEKTY

Pri metóde ohraničujúcich objektov sa zrýchlenie dosahuje výpočtom nad geometricky jednoduchším ohraničujúcim objektom oproti zložitejšiemu pôvodnému objektu. Ak je viditeľný ohraničujúci objekt, je viditeľný aj pôvodný ohraničený objekt. Táto metóda sa používa aj v prípade detekcie kolízií.



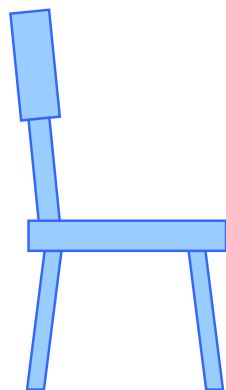
OHRANIČUJÚCE OBJEKTY

- Hierarchia ohraničujúcich gulí (BSH)
- Osovo - orientovaný kváder (AABB)
- Objektovo - orientovaný kváder (OBB)
- Konvexná obálka (Convex Hull)

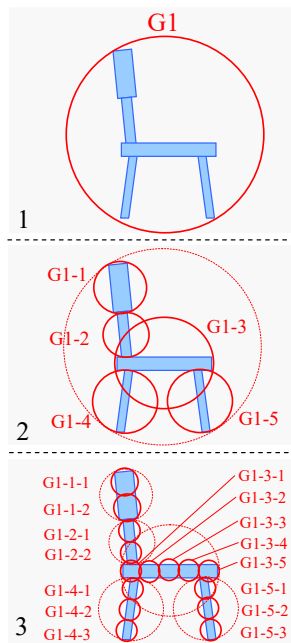


OHRANIČUJÚCE OBJEKTY – KOLÍZIE

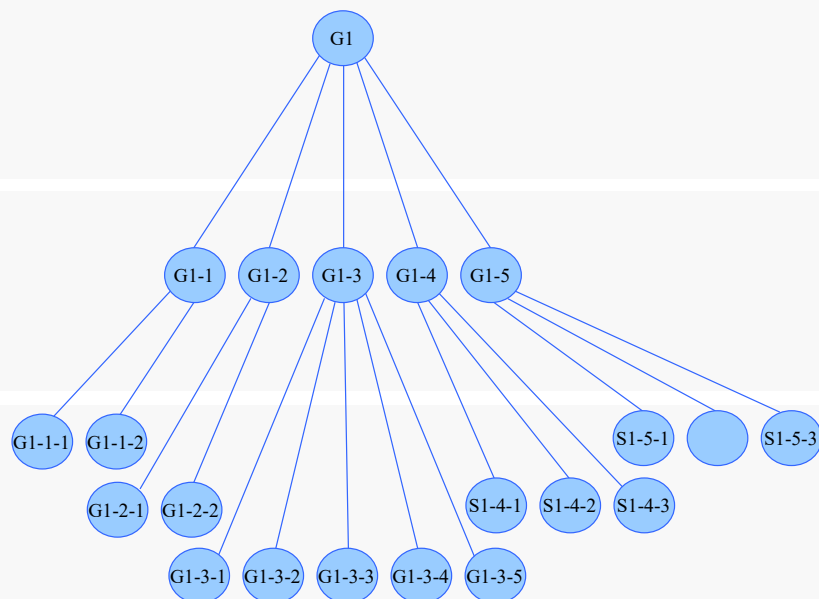
PRÍKLAD RIEŠENIA KOLÍZIE POMOCOU HIERARCHIE GULÍ V PROJEKCII 2D



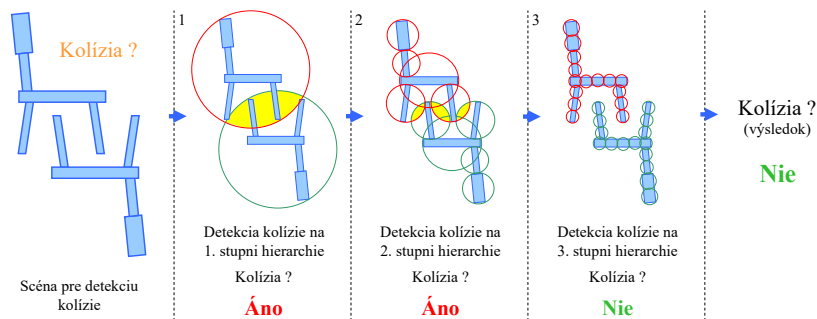
Objekt



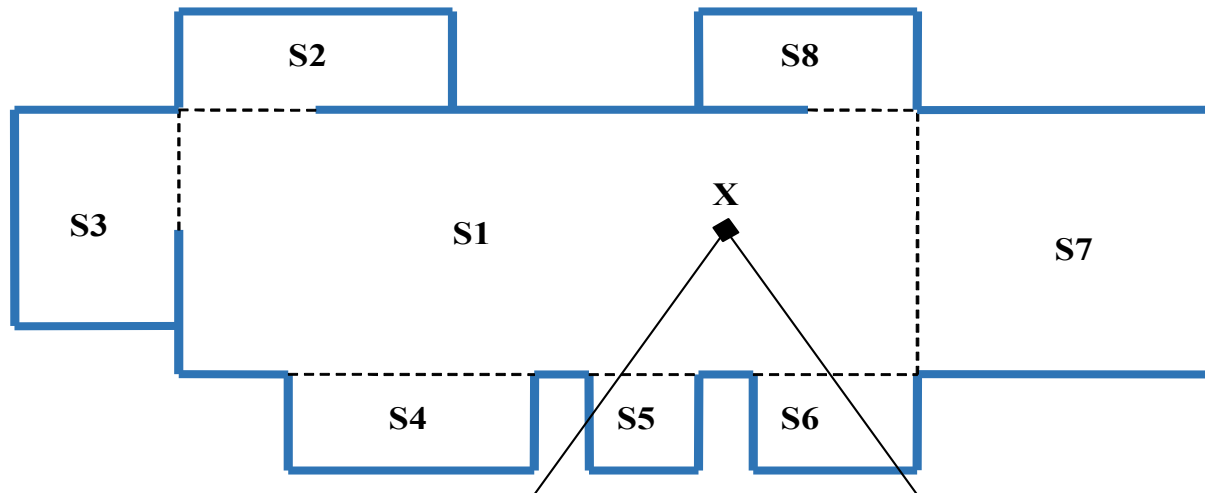
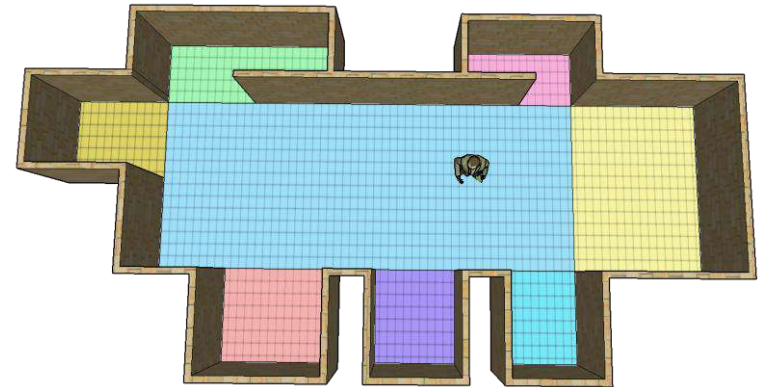
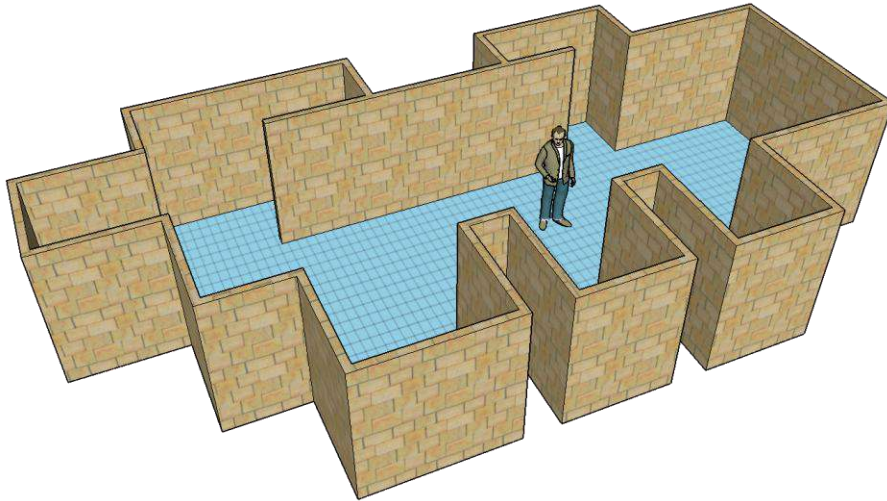
Úrovne hierarchie



Strom hierarchie

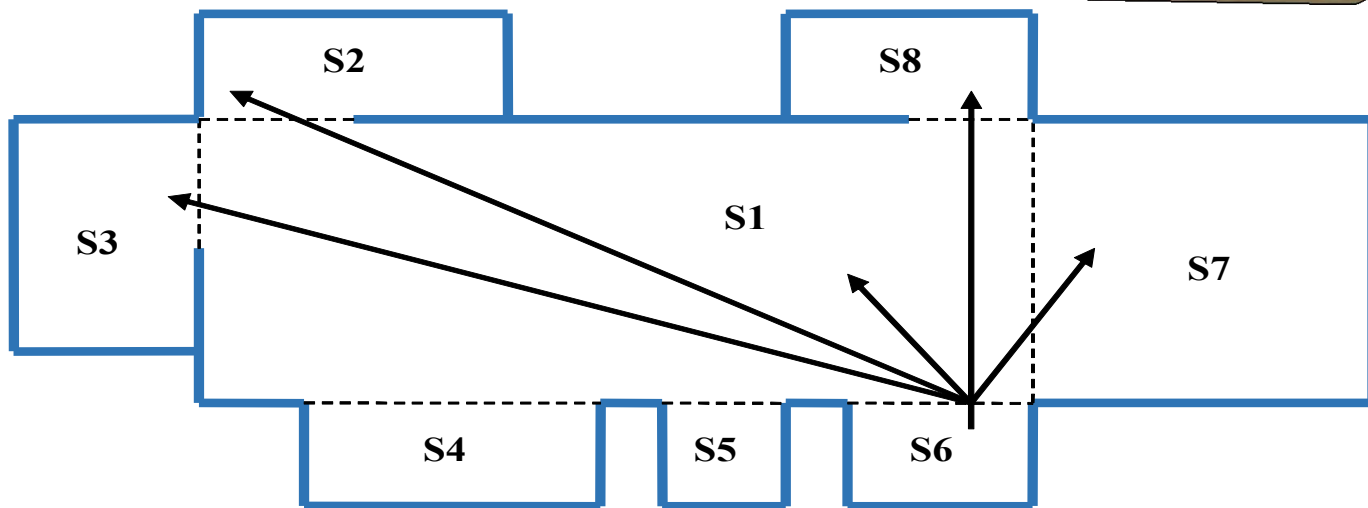
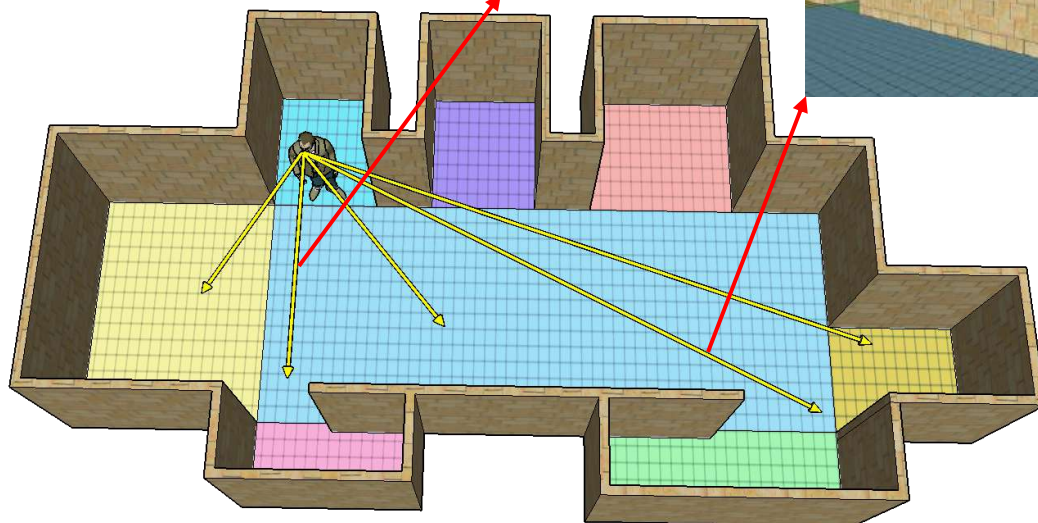
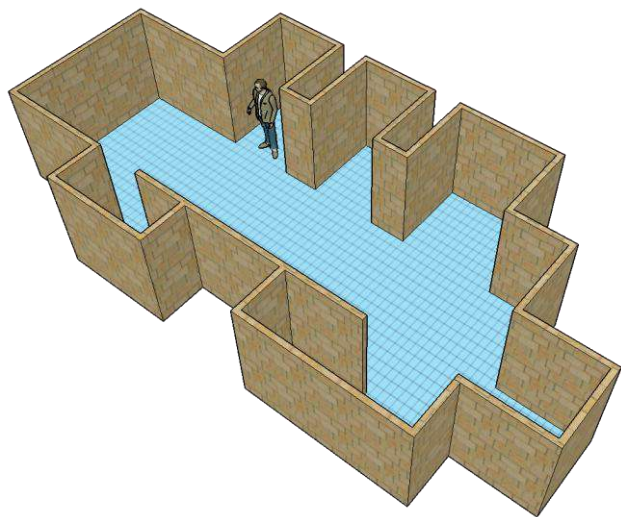


SEKTOROVANIE



POTENCIÁL VIDITEĽNOSTI

(POTENTIALLY VISIBLE SET, PVS)



S-BUFFER (SPAN BUFFER)

je jednou z techník, ako zabezpečiť neprekresľovanie už vykreslených polygónov v rámci SSZ, jedná sa teda o algoritmus, ktorý pracuje prioritne v 2D. Ak v 3D, potom sa ukladá hĺbka iba na začiatku a na konci spanu a medzitým sa interpoluje.

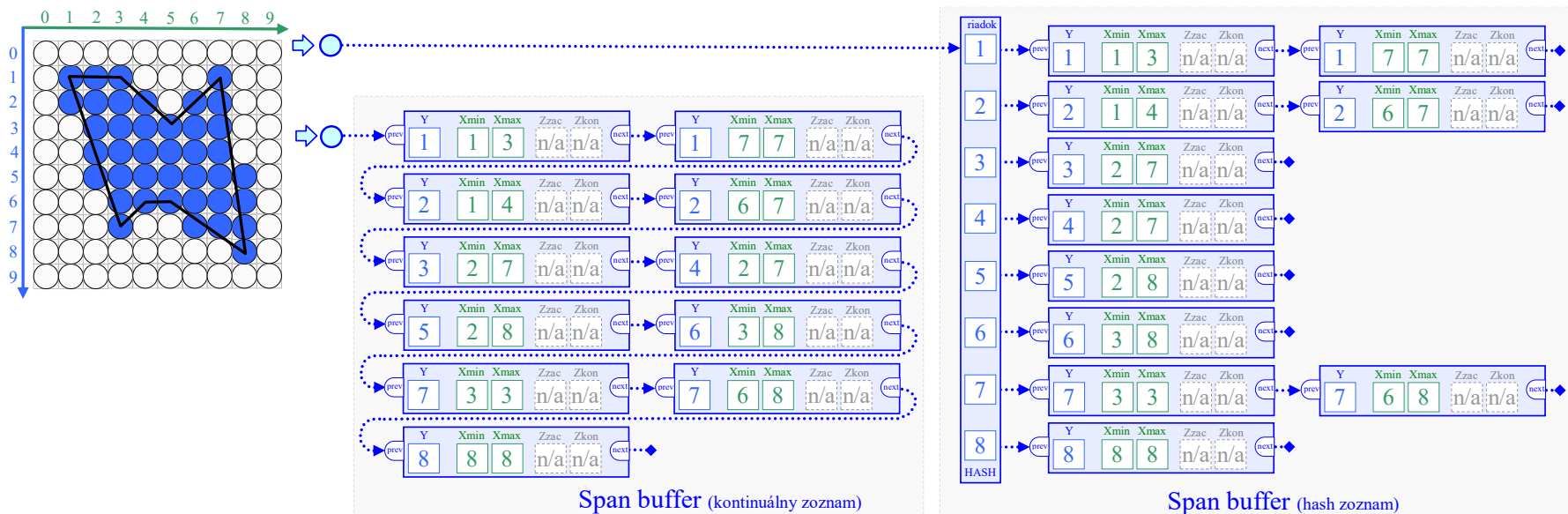
Jeho použitie sa teda obmedzuje na použitie v systémoch, ktoré kreslia scénu spredu dozadu.



S-BUFFER

ÚDAJOVÉ ŠTRUKTÚRY

Údajové štruktúry, ktoré sa používajú pri S-buffri sú rôzne. Často sa však používa najmä spojkový zoznam. Ide teda o spojkový zoznam častí/úsekov riadkov (angl. span), ktoré sú už vykreslené. Takýto zoznam je buď zostavený *kontinuálne* alebo *osobitne pre každý riadok* obrazovky a smerníky na prvý span na každom riadku sú uložené v hašovacej tabuľke.



S-BUFFER

SPRACOVANIE

Pri kreslení polygónu (mnohouholníka) sa musí každý *span* podrobiť testu, či nie je prekrytý iným, už nakresleným spanom. Ak je prekrytý, t.j. leží celý v rozmedzí x_0 a x_1 nejakého spanu v danom riadku, tak sa nekreslí. V opačnom prípade je potrebné počítat' prieniky spanov a vykresliť len tie časti, ktoré sú ešte nevykreslené, t.j. postupne zaplňovať diery v S-buffri. Rutina na vkladanie spanu nie je vôbec jednoduchá a v značnej miere na jej efektívnosti závisí aj výkon vizualizačného systému.

S-BUFFER VS. Z-BUFFER

Span-buffer		Z-buffer
• namiesto ukladania hĺbky pre každý pixel ukladá viditeľné horizontálne úseky (spany), hĺbka sa ukladá iba na začiatku a na konci spanu a medzitým sa interpoluje • je alternatívna technika k <i>Z-bufferu</i> • pri vykresľovaní sa kontroluje, či nový span pokrýva (podľa hĺbky) existujúci, a podľa toho sa aktualizuje • potrebuje menej pamäte (pracuje s úsekmi, nie s každým pixelom), efektívnejšie pre scény s veľkými polygónmi • má zložitejšiu implementáciu, horšie pre veľmi fragmentované scény • rieši viditeľnosť po častiach (úsekoch), čím je potrebný nižší počet operácií a menšia pamäť	vs. ↔	• ukladá hĺbku (Z hodnotu) pre každý pixel na obrazovke • je alternatívna technika k <i>Span-bufferu</i> • pri vykresľovaní každého bodu kontroluje, či je bližšie k pozorovateľovi než aktuálna hodnota v Z-bufferi • má veľkú pamäťovú náročnosť (pre každý pixel) • má jednoduchú implementáciu, funguje pre ľubovoľné scény • rieši viditeľnosť pixel po pixeli, čím je potrebný väčší počet operácií a väčšia pamäť

Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

VYPLŇOVANIE A TEXTÚROVANIE

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 08

© 2025

VRSTVY VIZUALIZAČNÉHO PROCESU

1. Definovanie/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



TYPY VYPLŇOVANIA

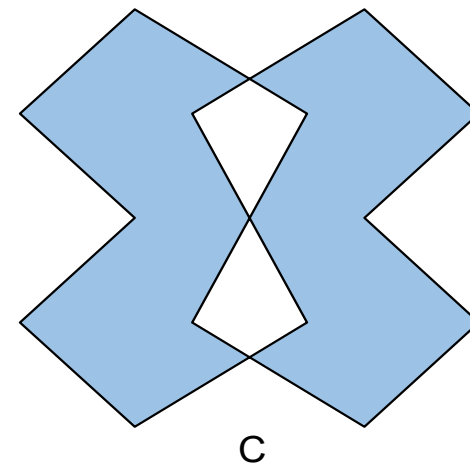
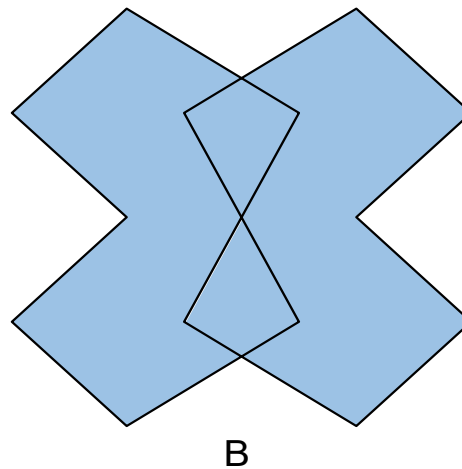
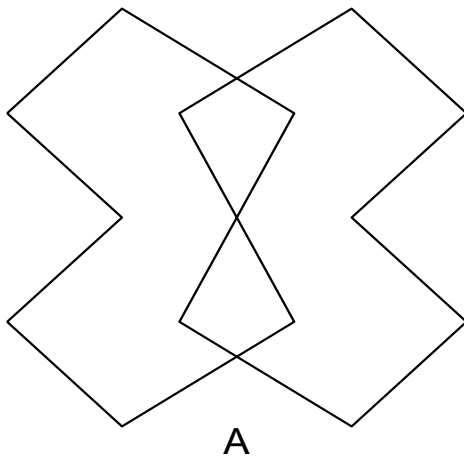
- vyplnenie oblasti jedinou farbou
- vyšrafovanie oblasti
- vyplnenie farebným vzorom (textúrovanie).



ROZDELENIE ALGORITMOV VYPLŇOVANIA

podľa toho akým spôsobom je zadaná hranica vyplňanej oblasti:

- hranica definovaná *geometricky*
- hranica *nakreslená na zobrazovači*



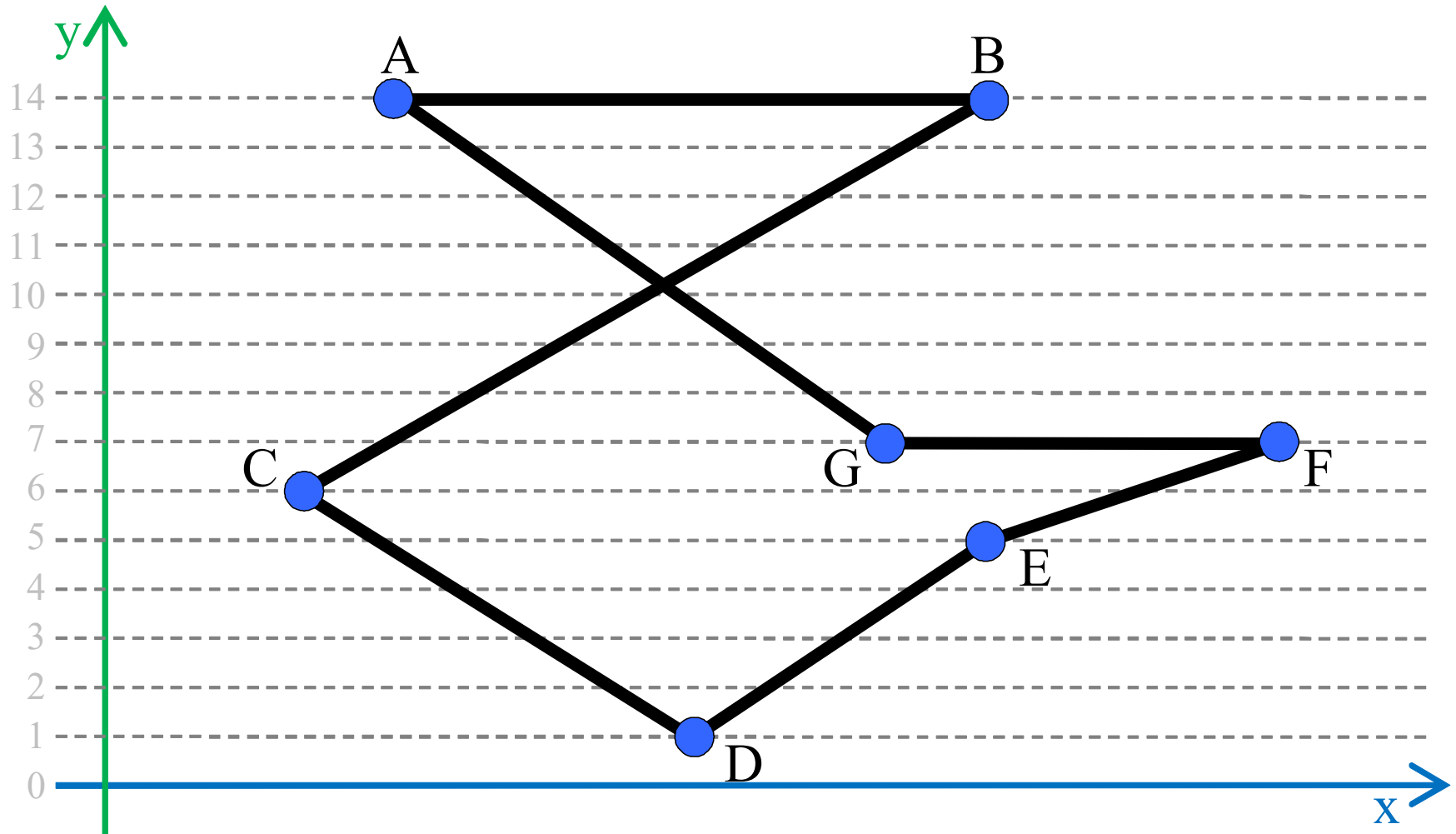
METÓDA RIADKOVÉHO ROZKLADU

SCANLINE

1. Postupuje sa od najvyššieho vrcholu oblasti k najnižšiemu, smerom zľava doprava v každom riadku.
2. Pre jednotlivé rozkladové riadky rovnobežné s osou X a s konštantnou súradnicou Y , klesajúcou s krokom -1 , sa nájdu priesečníky s hranicami vyplňanej oblasti.
3. Vo výslednom zozname sú usporiadané priesečníky zľava doprava (podobne ako span-buffer) a vyfarbené úseky medzi nepárnymi a párnymi priesečníkmi. Preto je nutné aby počet priesečníkov bolo párne číslo.

METÓDA RIADKOVÉHO ROZKLADU

SCANLINE

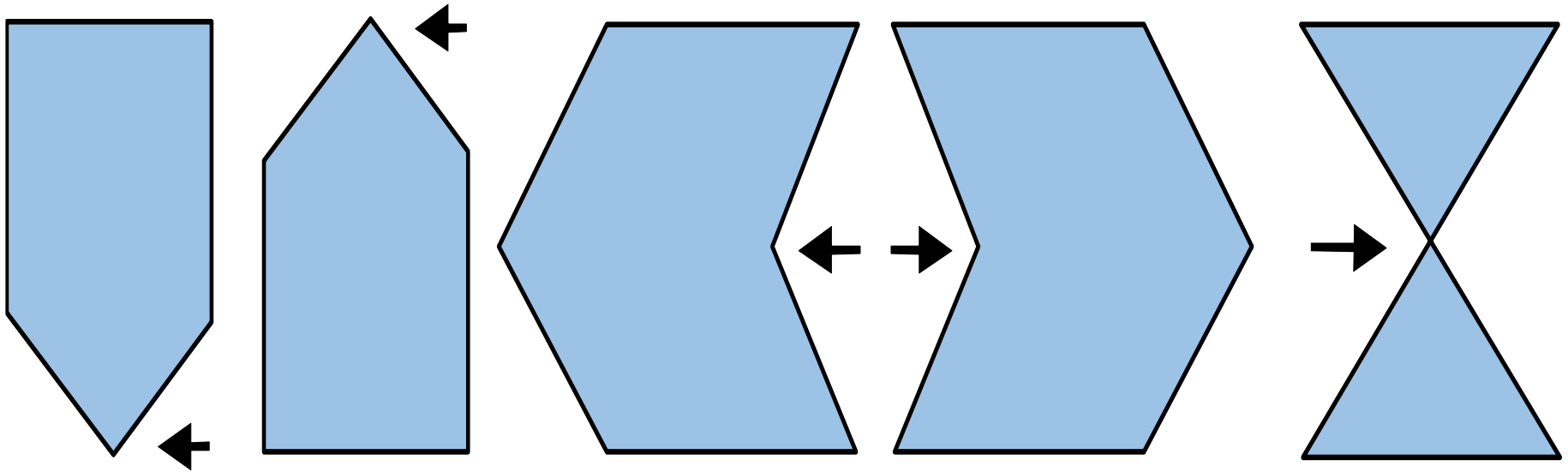


NÁJDENIE PRIESEČNÍKOV PRI METÓDE RIADKOVÉHO ROZKLADU

$$x = a_x + \frac{b_x - a_x}{b_y - a_y} \cdot (c_y - a_y)$$

$$y = c_2$$

EXTRÉMY PRI METÓDE RIADKOVÉHO ROZKLADU



INVERZNÉ VYPLŇOVANIE

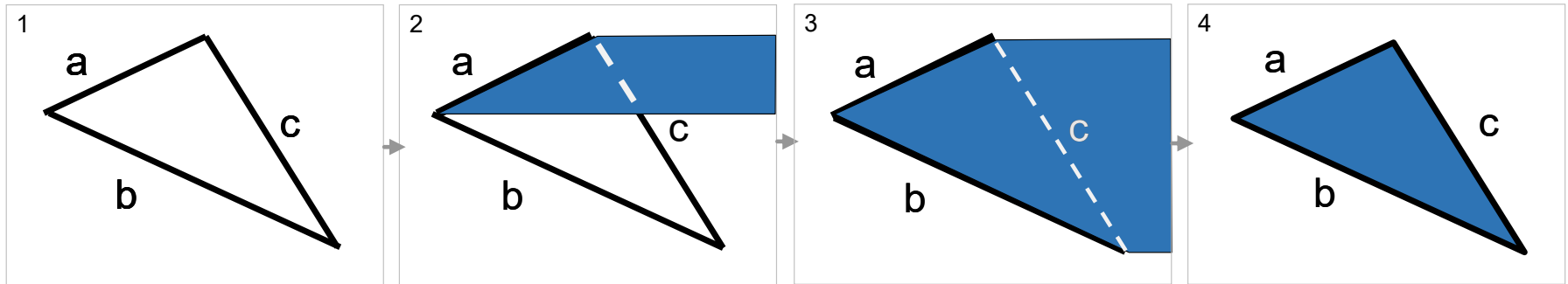
Využíva metódu XOR ($0 \vee 0 = 1 \vee 1 = 0$, $0 \vee 1 = 1 \vee 0 = 1$)

Dva typy:

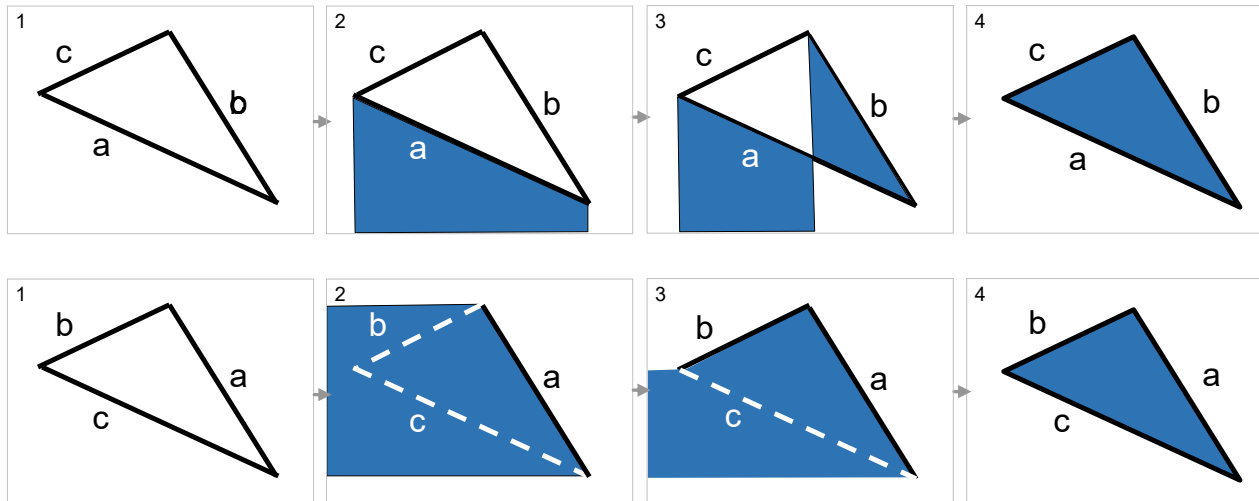
- vodorovné (riadkové) vyplňovanie
- zvislé (plotové) vyplňovanie



INVERZNÉ VYPLŇOVANIE



alternatívy:



VYPLŇOVANIE SPEKTROM (2 SPÔSOBY)

Prvý spôsob:

výplne spektrom využíva metódu riadkového rozkladu popísanú skôr.

- Na začiatku sa musí otočiť mnohouholník tak, aby smer vykresľovania priamok bol rovnobežný s osou x , t.j. o uhol β a metódou riadkového rozkladu sa nájdu priesečníky.
- Pri vykresľovaní úsekov medzi priesečníkmi sa musia úseky spätne otočiť o uhol β .
- Rozdiel oproti riadkovej výplni je v tom, že oblasť nie je vyplňaná jednou farbou, ale farba postupne plynule prechádza z prvej farby do druhej.

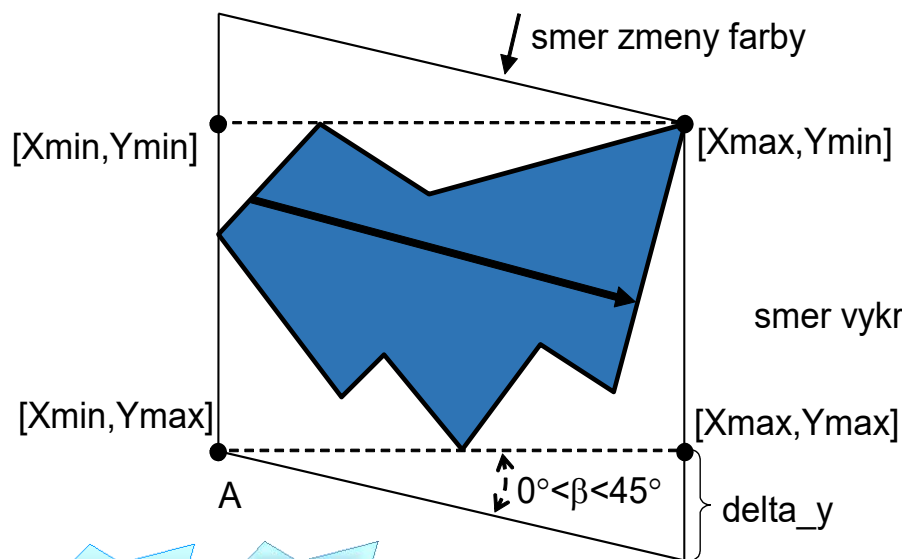
VYPLŇOVANIE SPEKTROM (2 SPÔSOBY)

Druhý spôsob:

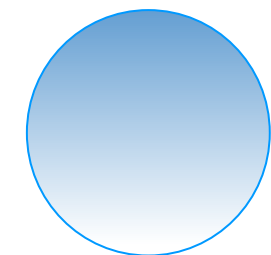
využíva možnosť nastavenia orezávacej oblasti, ktorú poskytuje napr. grafické rozhranie MS Windows.

- Na začiatku sa nastaví orezávacia oblasť na celú oblasť mnohoúhelníka a vypočítajú sa súradnice $Xmin$, $Xmax$, $Ymin$, $Ymax$.
- Podľa veľkosti uhla β je vyplnený príslušný rovnobežník a orezávacia oblasť zabezpečí vyplnenie len v oblasti mnohoúhelníka.

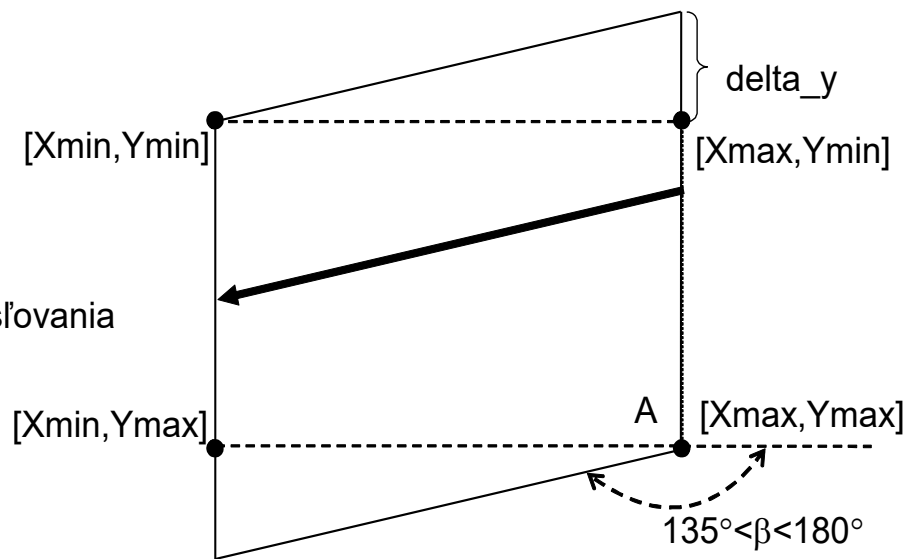
VYPLŇOVANIE SPEKTROM



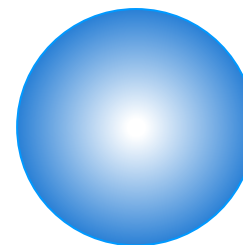
a



Lineárna zmena farby



b



Radiálna zmena farby

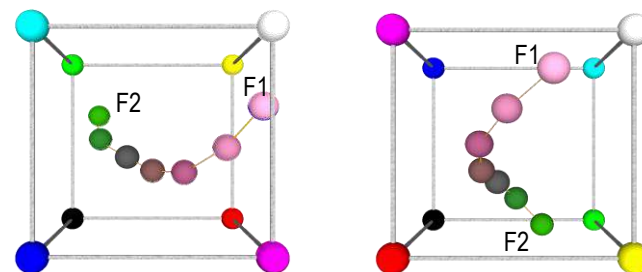
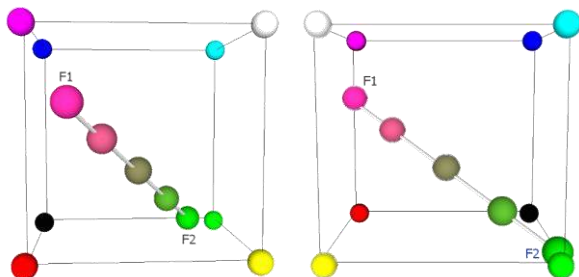
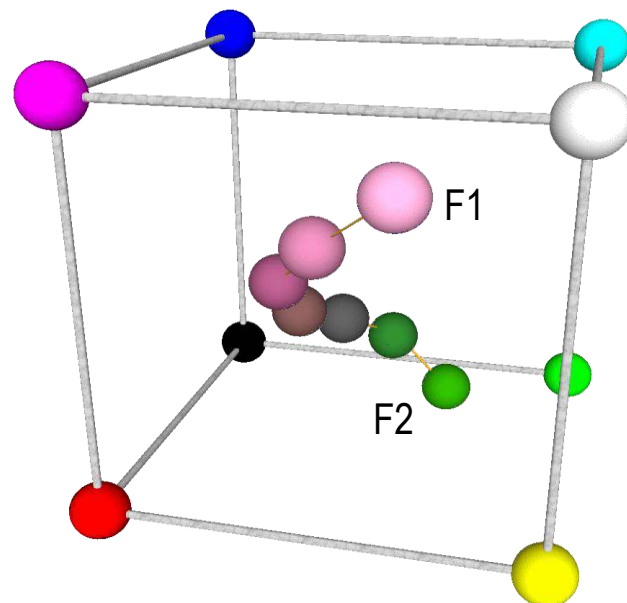
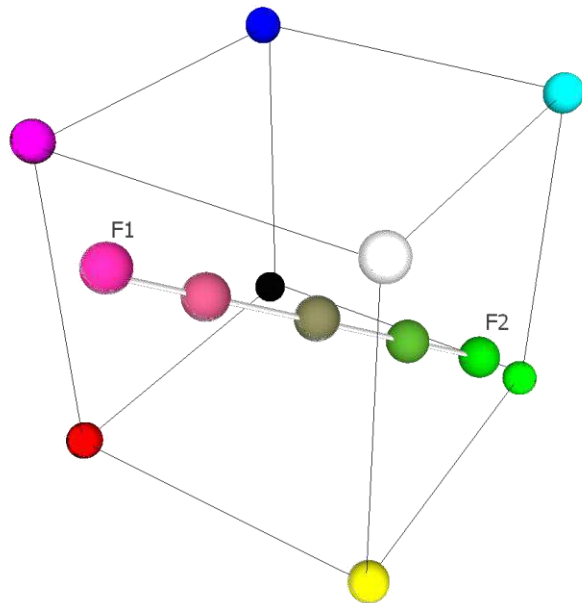
VYPLŇOVANIE SPEKTROM

Plynulý prechod farieb:

- Zistenie rozdielu jednotlivých zložiek (napr. RGB) prvej a druhej farby.
- Vypočítanie počtu úsečiek v rovnobežníku.
- Vypočítanie prírastkov jednotlivých zložiek ako pomer rozdielu zložiek prvej a druhej farby a počtu úsečiek, napr. pre zložku R
- alebo prechod napr. v RGB modeli

VYPLŇOVANIE SPEKTROM (PRECHOD V RGB MODELI)

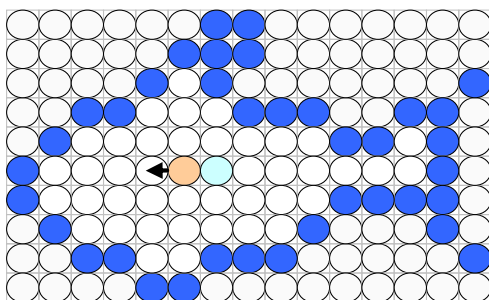
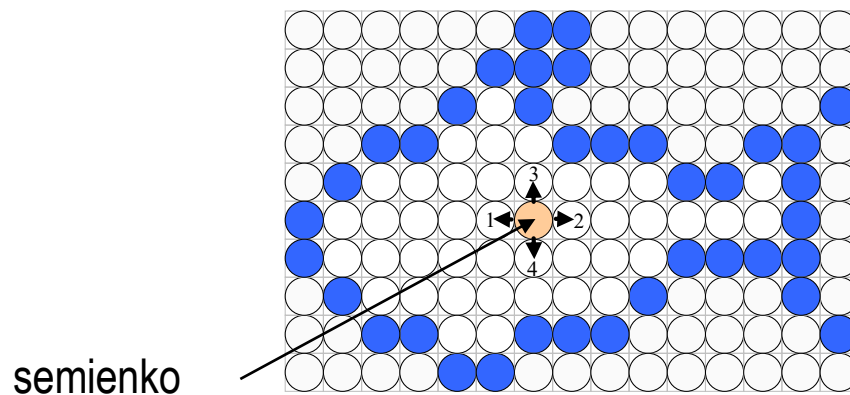
Ekvivalent alfa miešania (lineárne aj nelineárne, RGB priestor)



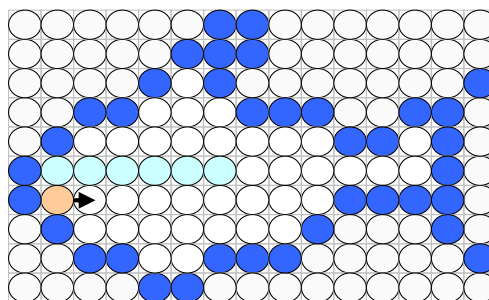
SEMIENKOVÉ VYPLŇOVANIE

```
SeedFill (x, y)
{
    SetPixel (sx, sy, BLUE);
    if (((GetPixel (sx-1, sy)) != HR_FARBA) &&
        ((GetPixel (sx-1, sy)) != BLUE)) SeedFill (sx-1, sy);
    if (((GetPixel (sx+1, sy)) != HR_FARBA) &&
        ((GetPixel (sx+1, sy)) != BLUE)) SeedFill (sx+1, sy);
    if (((GetPixel (sx, sy-1)) != HR_FARBA) &&
        ((GetPixel (sx, sy-1)) != BLUE)) SeedFill (sx, sy-1);
    if (((GetPixel (sx, sy+1)) != HR_FARBA) &&
        ((GetPixel (sx, sy+1)) != BLUE)) SeedFill (sx, sy+1);
}
```

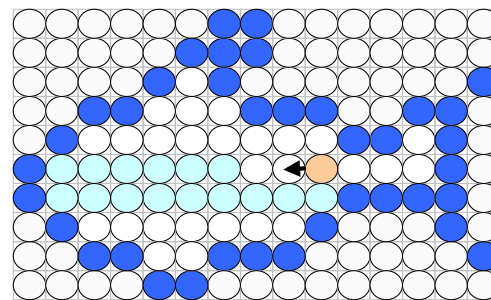
SEMIENKOVÉ VYPLŇOVANIE



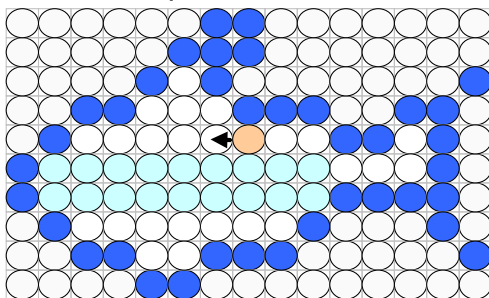
po 1. kroku



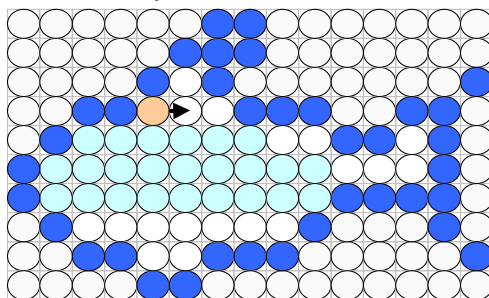
po 6. kroku



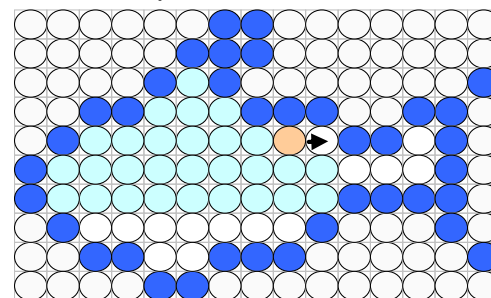
po 15. kroku



po 18. kroku

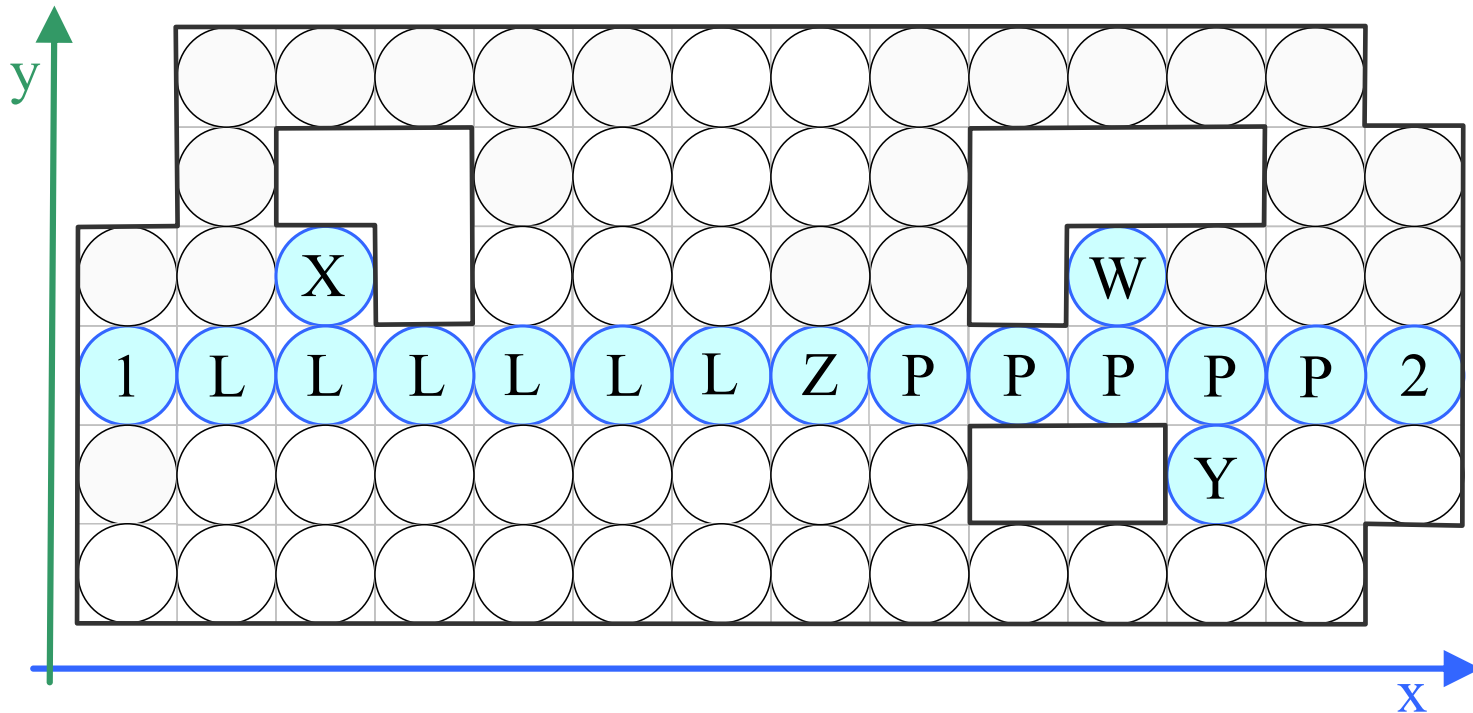


po 24. kroku



po 28. kroku

NEREKURZÍVNE SEMIENKOVÉ VYPLŇOVANIE



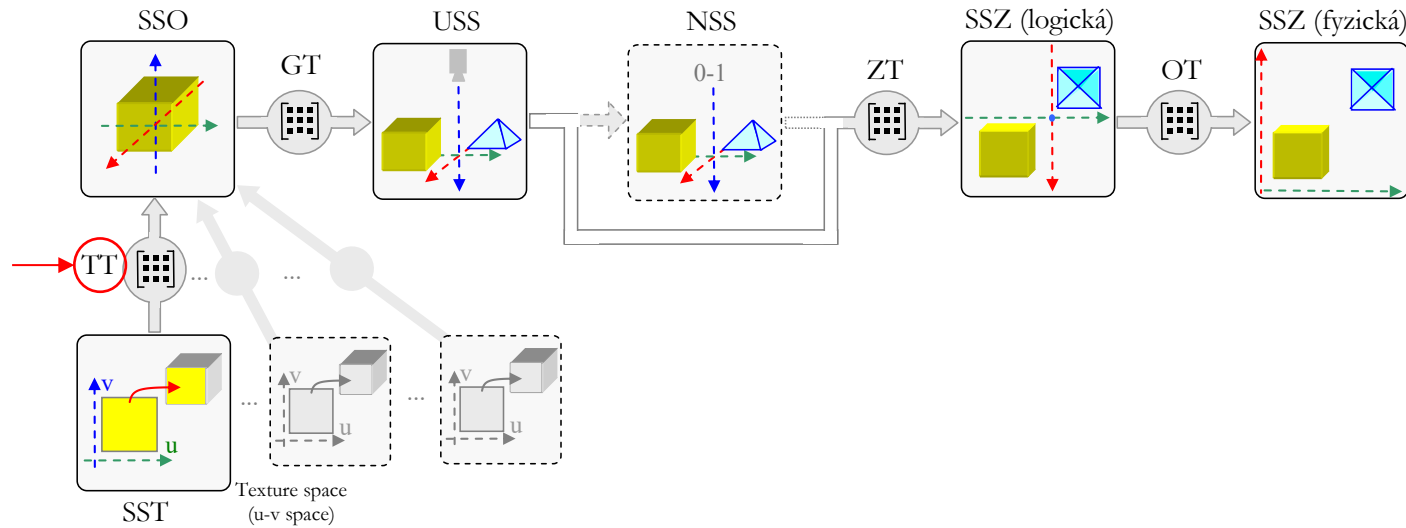
TEXTÚROVANIE

Proces (Textúrovacia transformácia, TT) nanášania obrazových vzoriek (textúr, tapiet) na povrch objektov za účelom získania vizuálneho dojmu, že objekt je z istého materiálu (napr. drevo, kameň, kov a pod.).

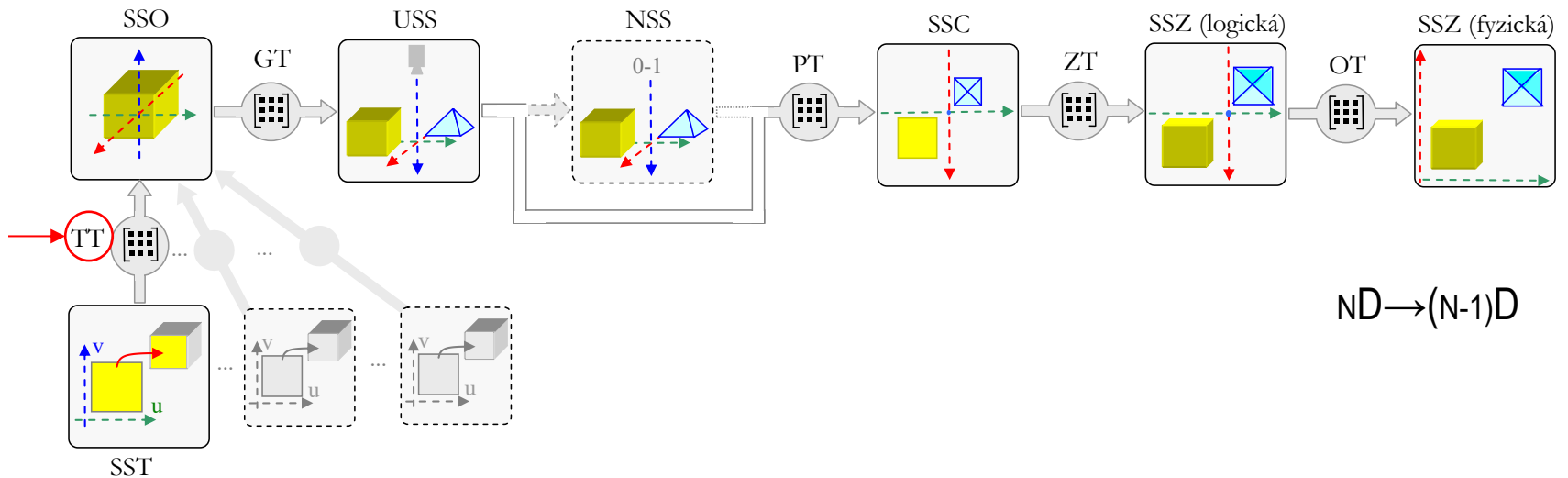
- Typy textúr podľa topologického rozmeru:
 - 1D, **2D**, 3D
- Typy textúr podľa spôsobu nanášania:
 - Statické
 - Dynamické
 - Procedurálne
 - Animačné



TEXTÚROVANIE A TRANSFORMAČNÉ REŽAZCE



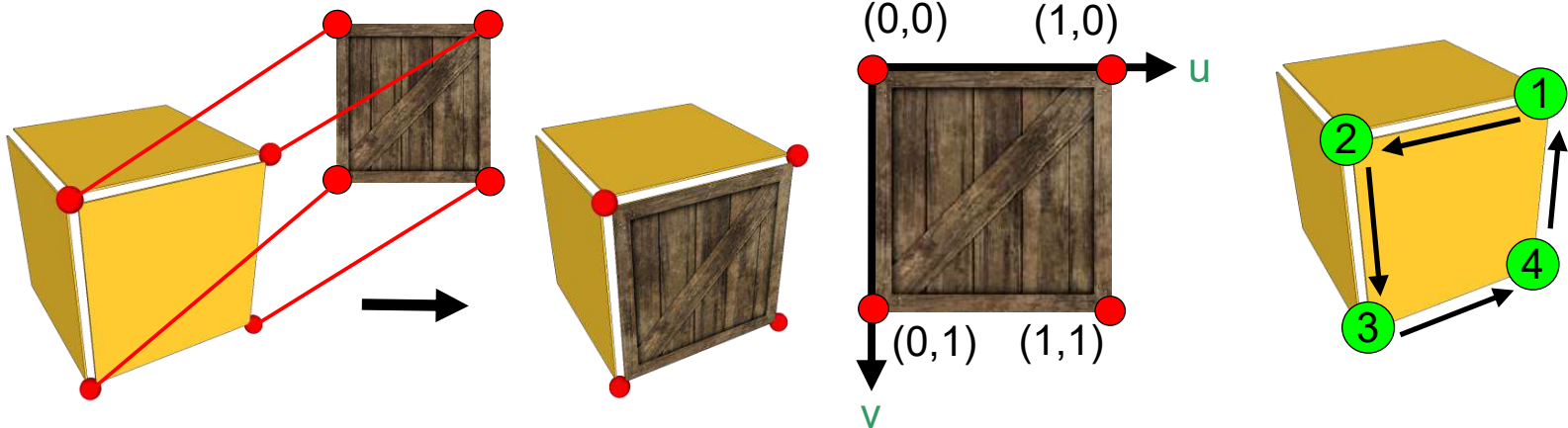
ND → ND



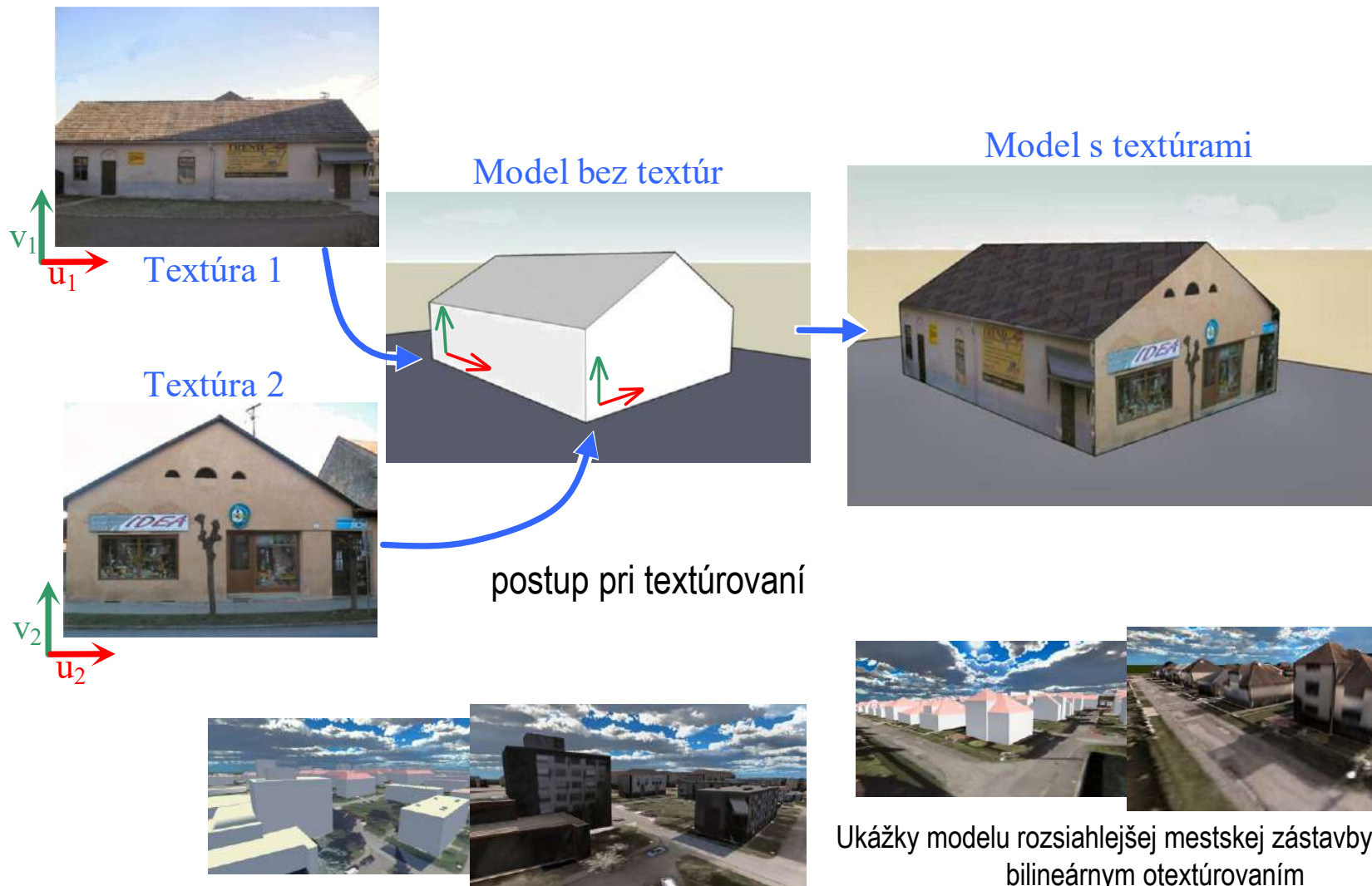
ND → (N-1)D

TEXTÚROVANIE (BILINEÁRNE, UV)

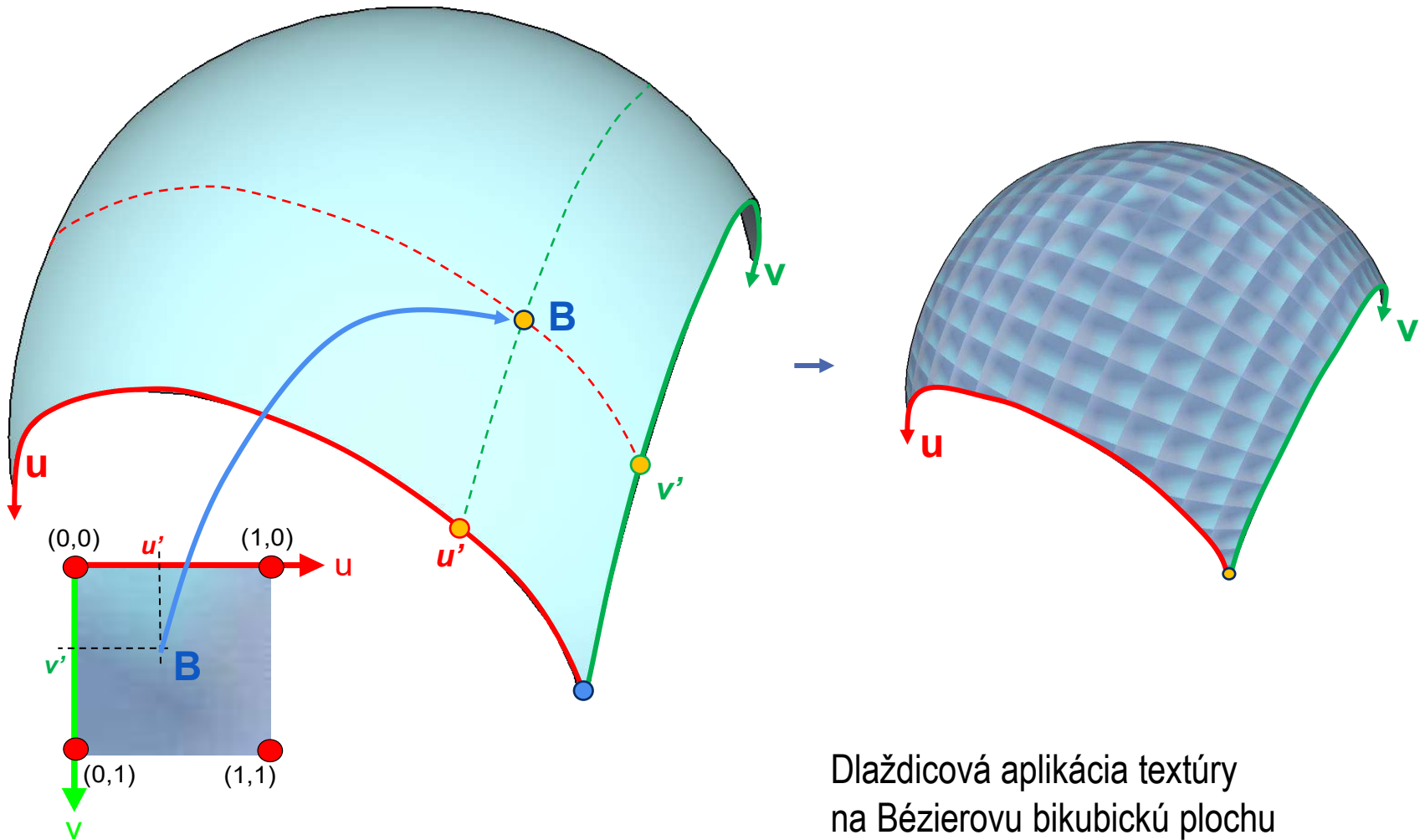
- Pri tomto procese nanášania obrazového formátu na plochy trojrozmerného objektu sa najčastejšie používa ako vzor obrázok (textúra).
- Obrazový formát (obrázok, textúra) využíva dvojrozmernú súradnicovú sústavu SST $[u, v]$, ktorej hodnoty súradníc u, v zodpovedajú jednotkovej miere obrazového formátu.
- Súradnica „ u “ zodpovedá x-ovej osi obrázka (textúry),
- súradnica „ v “ zodpovedá y-ovej osi textúry.
- Jedna jednotka = dĺžka/šírka obrázku textúry. (u/v)



TEXTÚROVANIE (BILINEÁRNE, UV)

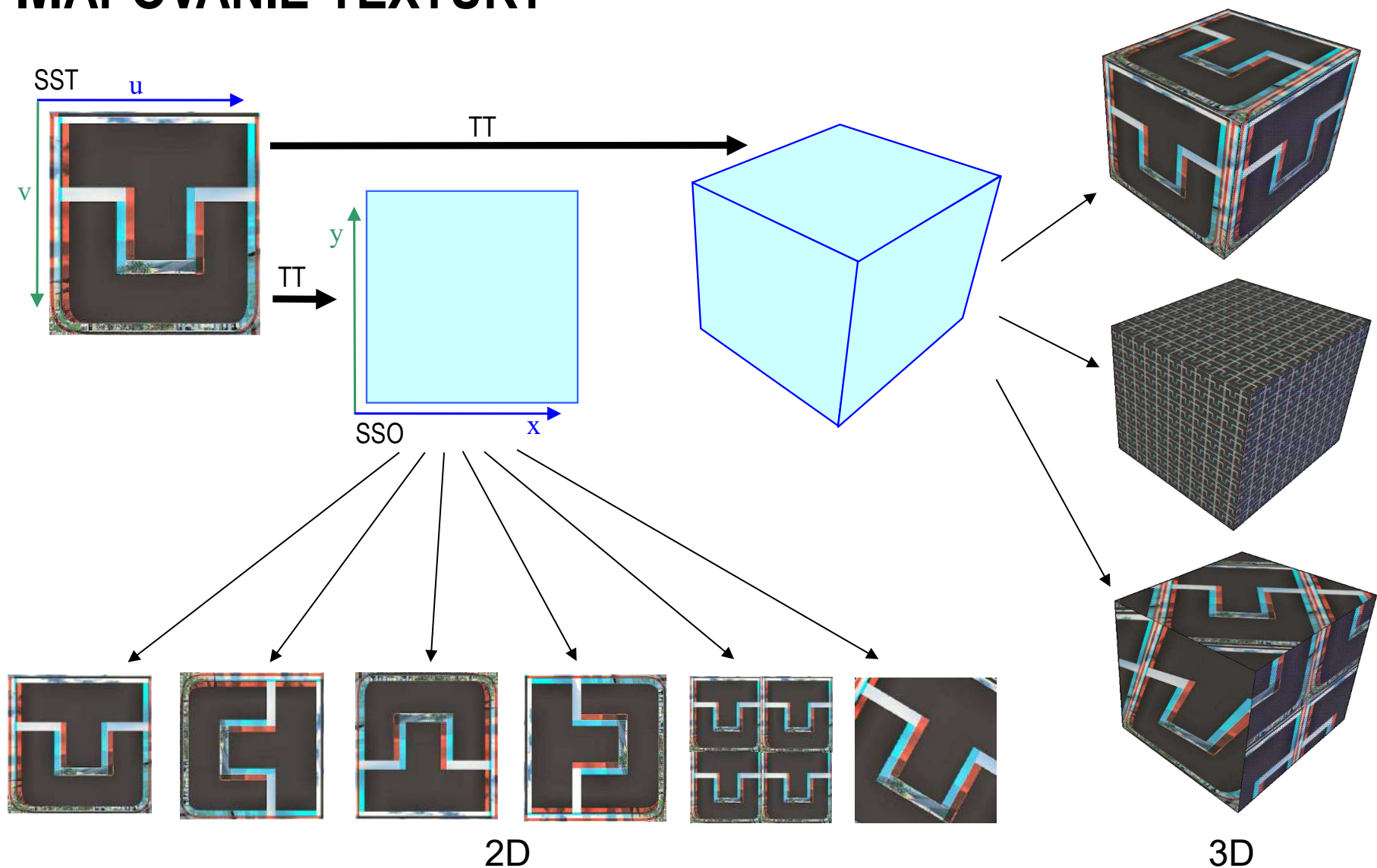


TEXTÚROVANIE (BILINEÁRNE, uv)

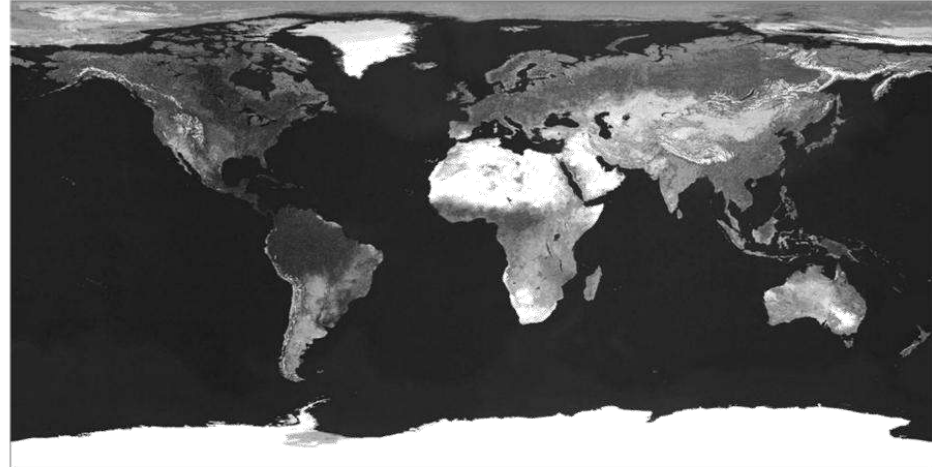


Dlaždicová aplikácia textúry
na Bézierovu bikubickú plochu

MAPOVANIE TEXTÚRY



BUMP MAPOVANIE TEXTÚRY



normal



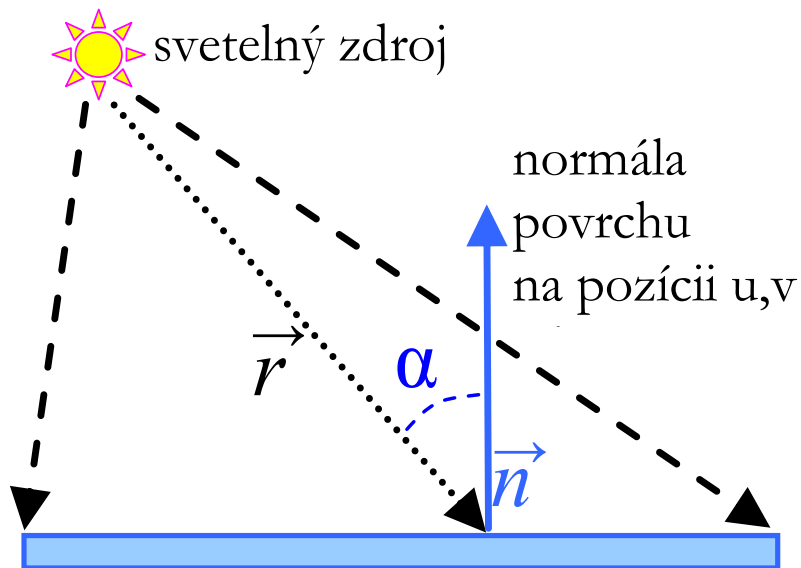
s tieňovaním



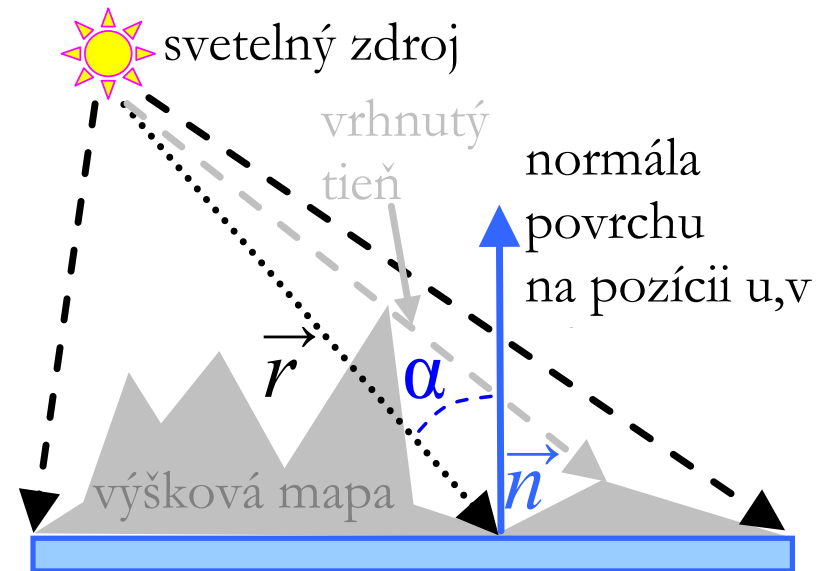
bump (s výškovou mapou)

IMPLEMENTÁCIA TIEŇOVANIA A OSVETĽOVANIA TEXTÚRY

$$\alpha = \arccos \left(\frac{r_x \cdot n_x + r_y \cdot n_y + r_z \cdot n_z}{\sqrt{r_x^2 + r_y^2 + r_z^2} \cdot \sqrt{n_x^2 + n_y^2 + n_z^2}} \right)$$



s tieňovaním

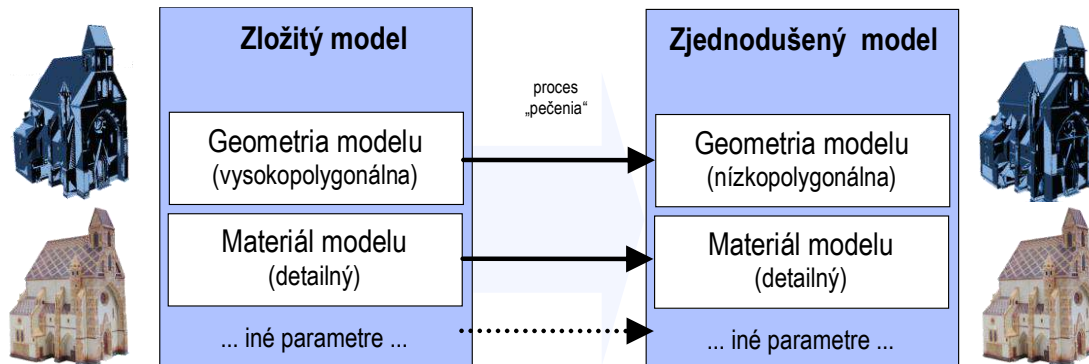


s výškovou mapou

upravuje sa farba resp. jas alebo saturácia bodov textúry, čím vzniká dojem tieňovania alebo hrboľatosti povrchu

„PEČENIE“ TEXTÚRY (TEXTURE BAKING)

- „Pečenie“ textúry je proces prenosu údajov o textúre (materiále) zo zložitejšieho (detailnejšieho, presnejšieho) 3D modelu na iný jednoduchší (menej presný, decimovaný) 3D modelu. Tento proces je možné zahrnúť medzi LOD techniky.
- Prenášané parametre/údaje pri tomto procese môžu byť napríklad: textúra, farba, osvetlenie, tieň, odrazy a podobne
- Vo všeobecnosti sa tento proces používa najmä pri snahe vziať model definovaný vysokým počtom polygónov (highpoly) a previesť ho na zjednodušený model s nízkym počtom polygónov (lowpoly). Najčastejšie dôvody použitia:
 - nízkopolygonálny model má menej údajov, a preto má menšiu veľkosť súboru ako vysokopolygonálny model.
 - mnohé aplikácie najmä herné, webové prehliadače a napr. virtuálno-reálné systémy často zápasia so spracovaním vysokopolygonálnych modelov, najmä ak je potrebné ich spracovávať v reálnom čase.



LOD TECHNIKY (LEVEL OF DETAIL) V PG

Level of Detail (LOD) je technika, ktorá optimalizuje výkon/výpočet tým, že znižuje komplexnosť objektov resp. ich parametrov, najčastejšie v závislosti od:

- *vzdialenosti* od kamery alebo
- od *uhla* alebo *smeru* pohľadu kamery (v rozdielnych uhloch pohľadu kamery vrátane *uhla zorného poľa*)

LOD sa implementuje/aktivuje:

- *diskrétne* – v skokových intervaloch vzdialenosti alebo uhla najčastejšie zamenením objektu/materiálu
- *kontinuálne* – dynamicky (priebežne) prepočítavané podľa vzdialenosti alebo uhla vrátane možnosti kontinuálneho prepočítavania zložitosti objektu/materiálu

Poznámka: rozšírené informácie napr. aj na <https://lodbook.com>

LOD TECHNIKY V PG - ZÁKLADNÉ

Medzi základné techniky LOD v počítačovej grafike patria najmä:

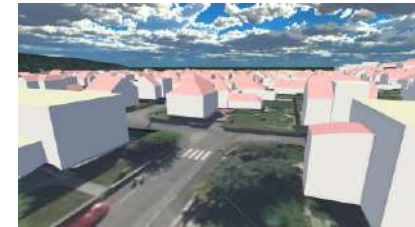
- **Geometrický LOD** (Geometric LOD, Tessellation LOD) - geometrická zložitosť objektu sa mení v závislosti od vzdialenosti alebo uhla kamery. Teselačný faktor sa mení podľa vzdialenosti od kamery, bližšie objekty majú viac detailov, vzdialené menej.
 - **Substitučný LOD** (Substitution LOD, Impostor LOD) - vzdialené 3D objekty sa nahradia 2D billboardmi (snímky objektu z rôznych uhlov). Je to typ geometrického LOD. Používa sa v hrách a simuláciách na zníženie počtu trojuholníkov pri zachovaní vizuálnej ilúzie.
 - **Obrazkový LOD** (Screen-space LOD, Resolution LOD) - určuje úroveň detailu podľa toho, koľko pixelov daný objekt zaberá v zornom poli oka na obrazovke.
 - **Hierarchický LOD** (Hierarchical LOD, Group LOD) - objekty v diaľke sa zhľukujú do väčších skupín a zobrazujú ako jeden jednoduchý objekt (napr. zhľuk stromov, vtákov a pod.).
 - **Terénny LOD** (Terrain LOD, geomipmapping) - je forma LOD optimalizácie terénnej grafiky tým, že sa rozdelí terén na dlaždice (tiles), z ktorých každá má svoju LOD úroveň (HLOD). Potom sú vzdialené terénne dlaždice zobrazované s menším detailom.
- **Materiálový/textúrovací LOD** (Material LOD) – kvalita/veľkosť mapovanej textúry/materiálu sa volí na základe vzdialenosti (aj napr. texture baking alebo mipmapping (použitie „pyramídy“ rôzne veľkých textúr)) vrátane zahrnutia PBR (Physically Based Rendering) parametrov objektu (napr. hrboľatosť, metalickosť a pod). Pri veľmi veľkej vzdialenosti postačuje použitie farby namiesto textúry.
- **Osvetľovací LOD** (Lighting LOD) – použitie rôzneho zložitého osvetľovacieho modelu (napr. Lambertov alebo Phongov) na základe vzdialenosti modelu vrátane možnosti použitia osvetľovacích máp (lightmaps) a obmedzenia vrhania tieňov.
- **Stereoskopický LOD** (Stereo alebo Dual-eye rendering LOD) – určuje použitie alebo nepoužitie dvoch kamier, ak je vzdialenosť väčšia ako stereoskopické rozlíšenie oka (cca 220m).
- **Tieňovací LOD** (Shading LOD) – rôzne tieňovanie (a tým potreby počítanie rôzneho počtu normál) podľa vzdialenosti vrátane modernej možnosti voľby jednoduchšej alebo zložitejšej/náročnejšej jednotky hardvérového shadera.
- **LOD zorného poľa** (Foveated Eye-tracked LOD) – kde je vykresľovanie zamerané na foveu (stred sietnice oka). Optimalizuje výkon tým, že berie ohľad na zorné pole oka a zobrazuje v strede obrazu vyššie detaily a smerom k periférnemu pohľadu sa detail znižuje.

LOD TECHNIKY V PG – ROZŠÍRENÉ A URÝCHĽOVACIE TECHNIKY

- **Zjednodušovanie siete** (Mesh simplification, Mesh decimation) s minimalizáciou vizuálnej straty - znižovanie počtu (redukcia) vrcholov a polygónov (najmä trojuholníkov) buď redukciou alebo zoskupovaním (agregáciou) veľmi blízkych (edge-collapse, vertex-clustering) vrcholov. Často sa používa minimalizácia kvadratickej chyby (quadric error metrics), kde každému vrcholu sa priradí kvadratická chyba (počíta sa pomocou matíc, ktoré reprezentujú vzdialenosť bodu od rovín definovaných polygónov) vyjadrujúca vzdialenosť od pôvodných plôch polygónov. Potom pri agregácii (zoskupení) sa nová pozícia volí tak, aby sa minimalizoval súčet jednotlivých chýb.
- **Použitie rôznych rozlíšení/detailizácie v rámci jednej scény** (multiresolution meshes) podľa pohľadu v reálnom čase a podľa blízkosti ku kamere.
- **Spojenie LOD a postupov z riešenia viditeľnosti**, kde sa nevykresľujú objekty, ktoré sú úplne zakryté inými objektmi z pohľadu kamery (occlusion culling). Znižuje sa tým počet objektov na spracovanie a zvyšuje sa tým výkon. Na detekciu viditeľnosti je možné požiť rôzne algoritmy (Z-buffer, Freeman-Louterlov algoritmus a pod.)
- **Použitie neurónových sietí na generovanie alebo optimalizáciu rôznych úrovní gemetrického detailu** objektov (Neural Geometric LOD). Používa sa SDF funkcia (Signed Distance Function), ktorá vracia pre každý bod v priestore kladnú hodnotu (ak je bod mimo vyšetrovaného objektu) alebo zápornú hodnotu (ak je bod vo vnútri objektu) alebo nulovú hodnotu (ak je bod na povrchu objektu) v spolupráci s neurónovou sieťou. Tým sa presne definuje povrch objektu a je možné použiť adaptívne oktantové stromy pre určenie LOD (často aj v reálnom čase).

LOD TECHNIKY MIMO PG

- Použitie v rámci GIS (Geografické informačné systémy) systémov, kde LOD označuje úroveň priblíženia mapy: napr. LOD 0 = celosvetový pohľad, LOD 23 = detailná ulica. Každá úroveň má definovaný rozsah mierky napr. 1:500 000 pre príslušné LOD.
- V stavebníctve a architektúre sa LOD často interpretuje ako Level of Development využívaný pri formáte BIM (Building Information Modeling):
 - LOD 100 - koncept, návrh, iba objemové tvary
 - LOD 200 - približná geometria, umiestnenie, okná, dvere
 - LOD 300 - presná geometria, rozmerové údaje, konštrukčné prvky
 - LOD 400 - stavebné, konštrukčné alebo výrobné detaily, montážne informácie, špecifikácia materiálu a konštrukčných prvkov
 - LOD 500 - finálny, skutočný stav po výstavbe



LOD 21 (LOD100 arch.) vektor a textúrovane)



LOD200 a 300 arch. vektor

(zdroj SvF TU Košice)

Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

TIEŇOVANIE A OSVETĽOVANIE

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 09

© 2025

VRSTVY VIZUALIZAČNÉHO PROCESU

1. Definovanie/spracovanie modelu (reprezentácia, súradnicové systémy)
2. Transformácie nad objektami
3. Riešenie viditeľnosti
4. Tieňovanie
5. Osvetľovanie
6. Realistické zobrazovanie
7. Kompozícia a Vykresľovanie



TIEŇOVANIE

Proces vplyvu svetelného zdroja na objekt (najčastejšie na jeho povrch).

Typy tieňovaní:

- Konštantné (flat)
- Gourardovo (interpolácia farby)
- Phongovo (interpolácia normály)

Podporné technológie:

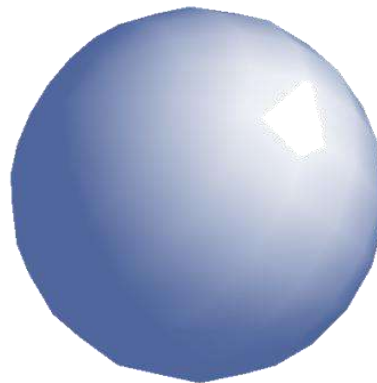
- Poltónovanie (halftoning)
- Rozptyľovanie (dithering)



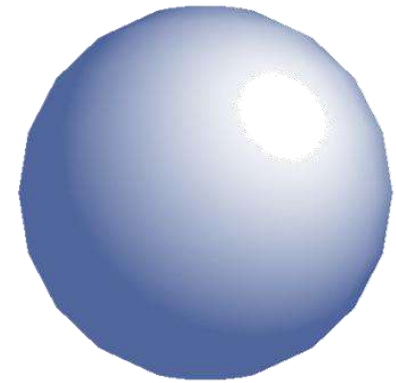
TIEŇOVANIE



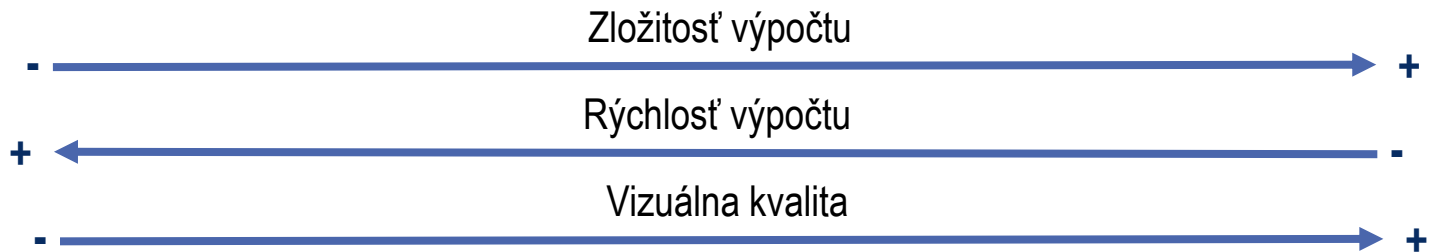
Konštantné
tieňovanie



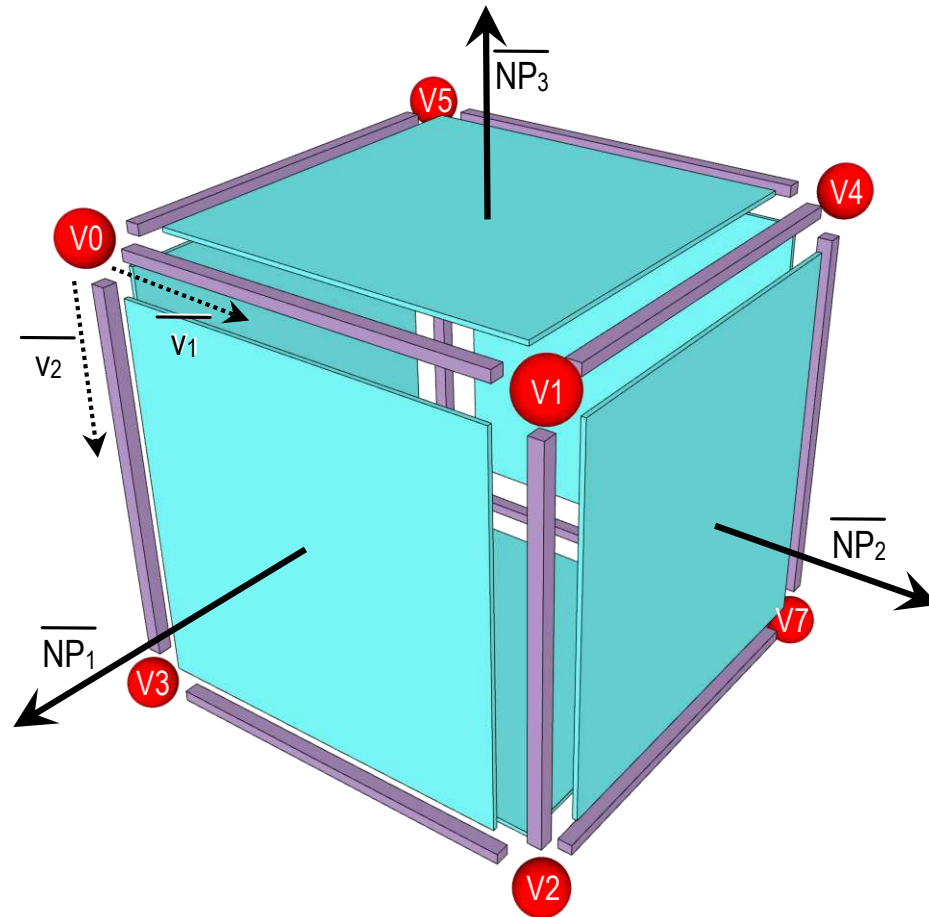
Gourardovo
tieňovanie



Phongovo
tieňovanie



KONŠTANTNÉ TIEŇOVANIE (FLAT)



$$\overline{NP_1} = \overline{v_1} \otimes \overline{v_2}$$

$\overline{v_1}$ a $\overline{v_2}$ sú smerové vektory hrán

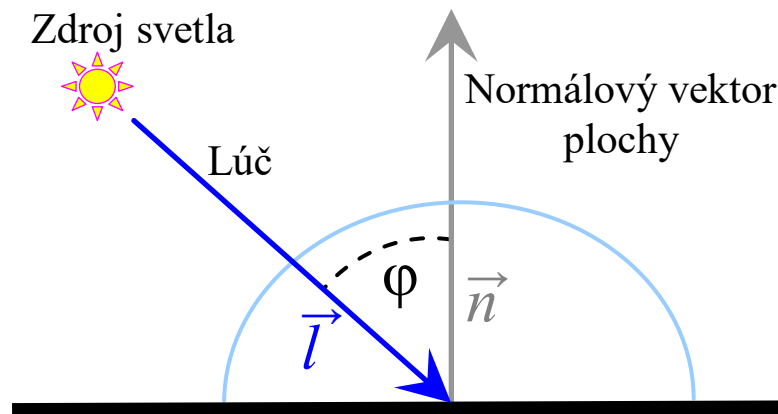
podmienka: rovnaká orientácia všetkých stien v scéne

určenie normál pri konštantnom tieňovaní

KONŠTANTNÉ TIEŇOVANIE (FLAT)

Intenzita tieňa je závislá od uhla φ medzi vektorom dopadu lúča $\vec{l}$ zo svetelného zdroja a normálou steny $\vec{n}$.

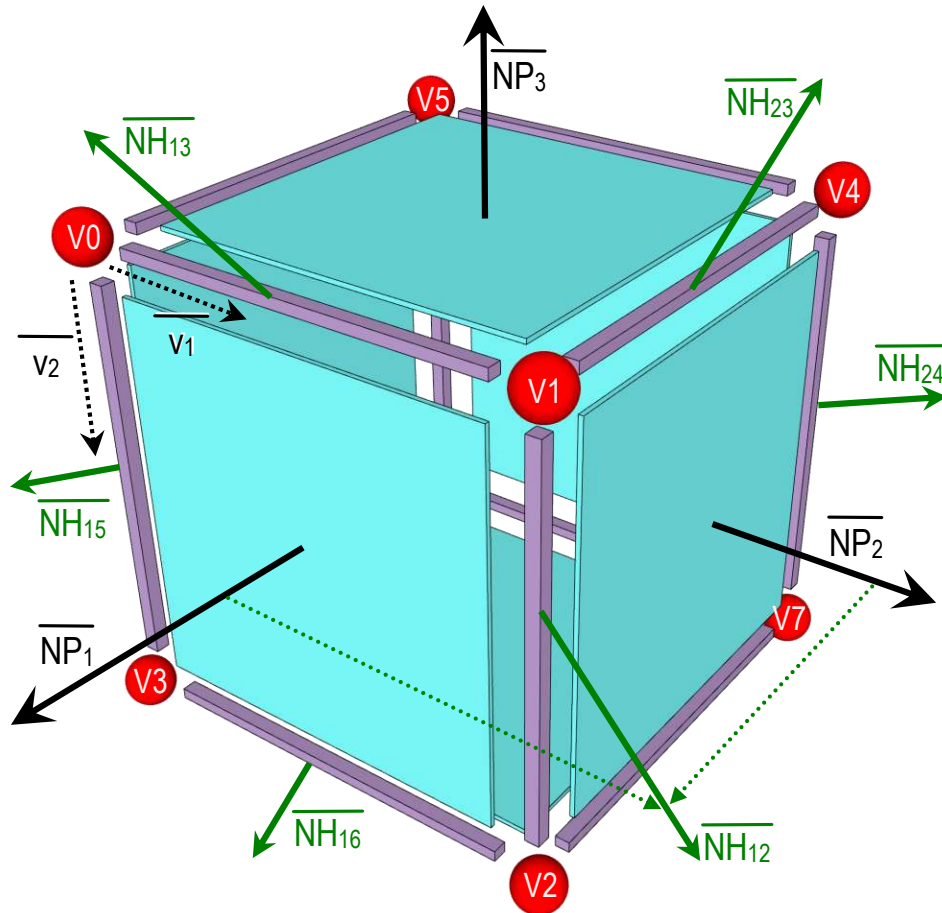
$$\varphi = \arccos \left(\frac{l_x \cdot n_x + l_y \cdot n_y + l_z \cdot n_z}{\sqrt{l_x^2 + l_y^2 + l_z^2} \cdot \sqrt{n_x^2 + n_y^2 + n_z^2}} \right)$$



GOURARDOVO TIEŇOVANIE (INTERPOLÁCIA FARBY)



Henri Gourard (1971)



$$\overline{NH_{12}} = \overline{NP_1} \oplus \overline{NP_2}$$

určenie normál pri Gourardovom tieňovaní

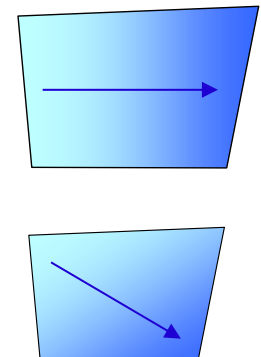
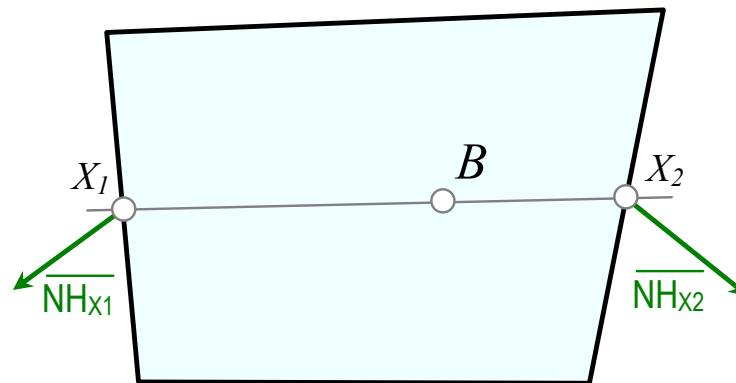
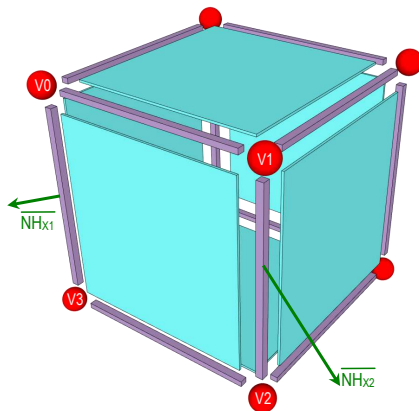
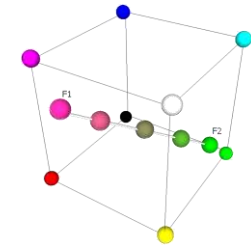
GOURARDOVO TIEŇOVANIE (INTERPOLÁCIA FARBY)

$$I_B = I_{X1} \cdot \frac{X_2 - X_B}{X_1 - X_2} + I_{X2} \cdot \frac{X_B - X_1}{X_2 - X_1}$$

I_{x1}, I_{x2} – intenzity (farby) v bodoch $x1$ a $x2$

I_B – výsledná interpolovaná intenzita (farba) v bode B

Interpolácia napr. v RGB priestore (ekv. alfa miešania)

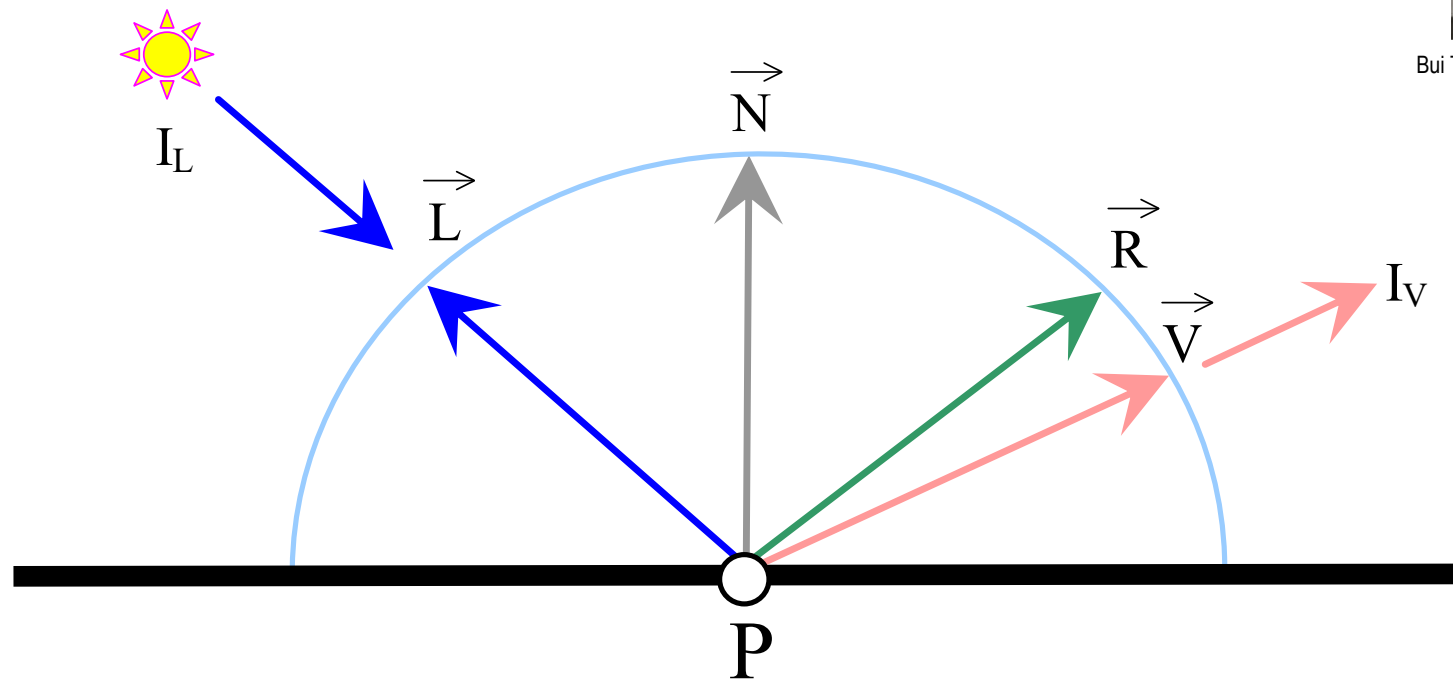


určenie farby interpoláciou pri Gourardovom tieňovaní

PHONGOVO TIEŇOVANIE (INTERPOLÁCIA NORMÁLY)

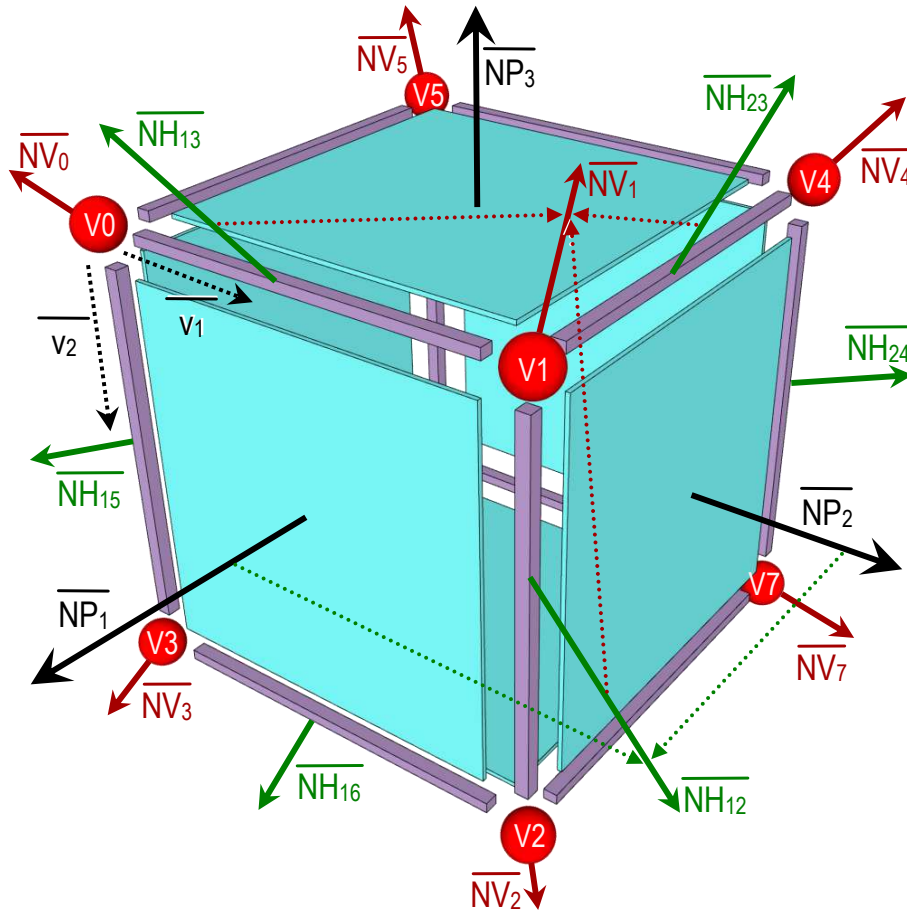


Bui Tuong Phong, (1942-1975)



Empirický osvetľovací model (Phong, 1975)

PHONGOVO TIEŇOVANIE (INTERPOLÁCIA NORMÁLY)



$$\overline{NV_1} = \overline{NH_{13}} \oplus \overline{NH_{12}} \oplus \overline{NH_{23}}$$

určenie normál pri Phongovom tieňovaní

PHONGOVO TIEŇOVANIE (INTERPOLÁCIA NORMÁLY)

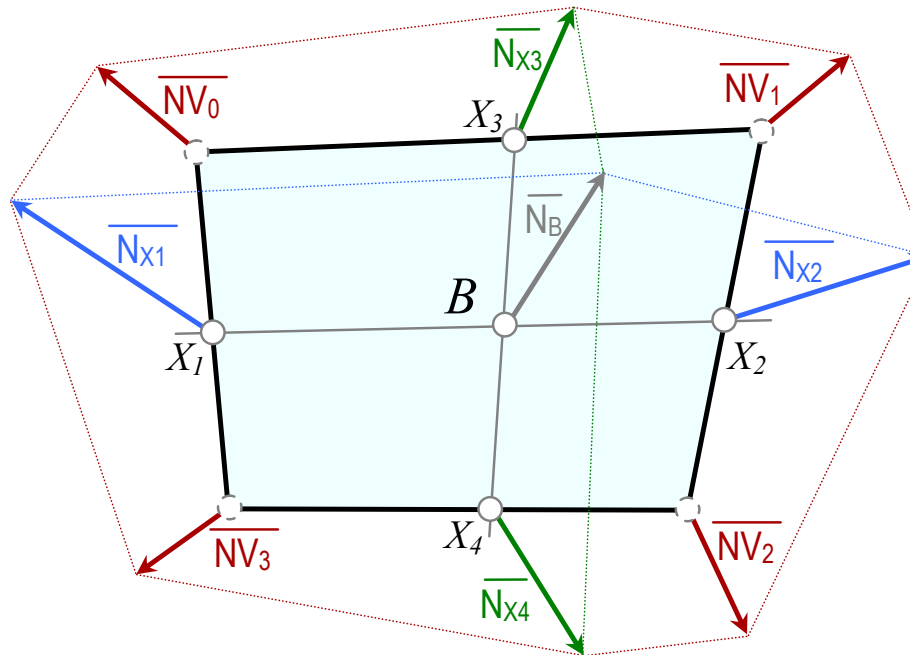
URČENIE NORMÁL

ekvivalentne pre $\overline{N_{X3}}$ a $\overline{N_{X4}}$

$$\overline{N_B} = \frac{\overline{N_{X1}} \oplus \overline{N_{X2}}}{2}$$

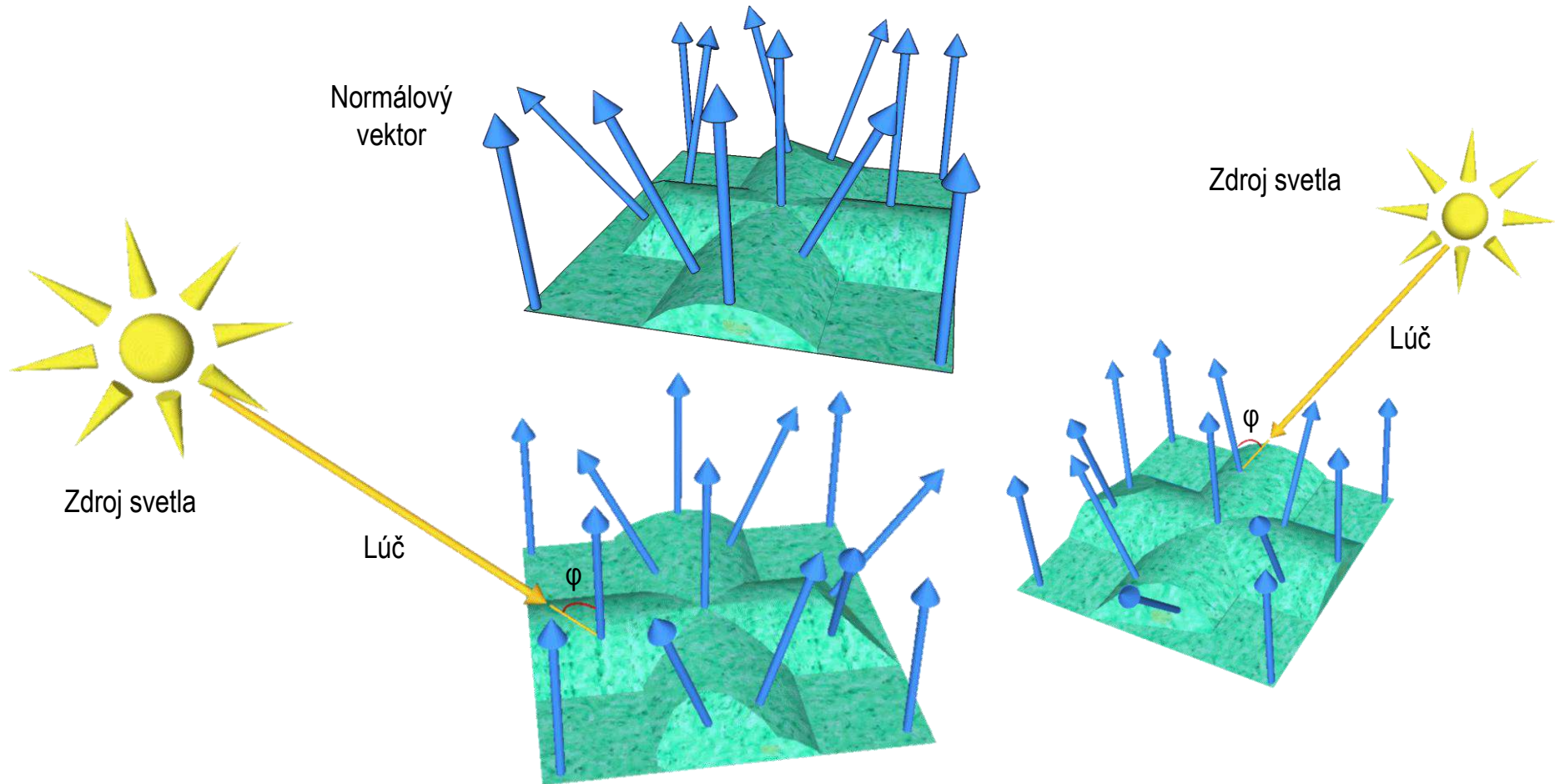
$$\overline{N_{X1}} = \frac{\overline{NV_0} \oplus \overline{NV_3}}{2}$$

$$\overline{N_{X2}} = \frac{\overline{NV_1} \oplus \overline{NV_2}}{2}$$

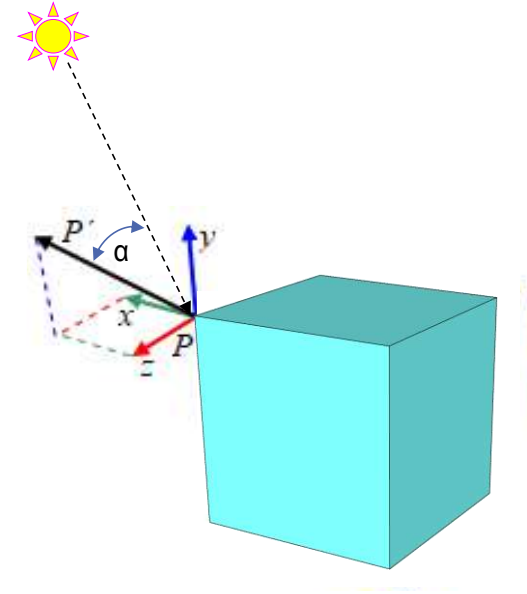
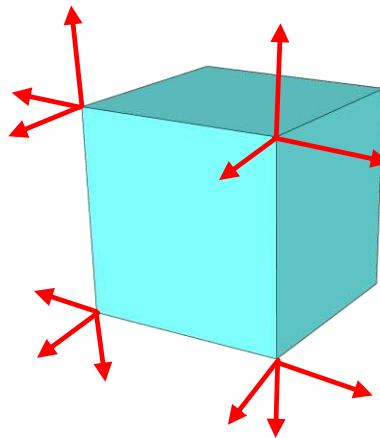
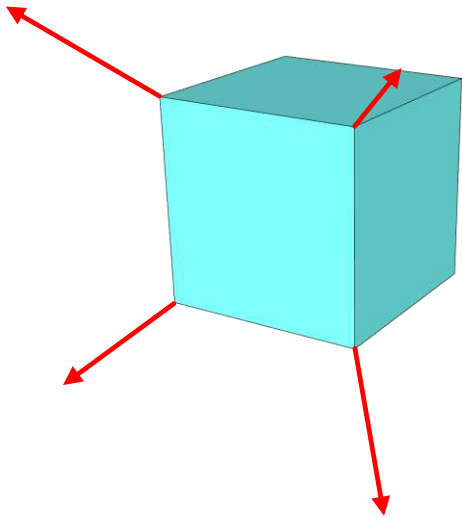


PHONGOVO TIEŇOVANIE (INTERPOLÁCIA NORMÁLY)

URČENIE NORMÁL PO INTERPOLÁCII (PRÍKLAD)



VRCHOLOVÉ TIEŇOVANIE



OSVETĽOVANIE

Proces vplyvu svetelného zdroja/zdrojov, materiálu a iných objektov na svoje okolie resp. na iné objekty – vrhanie tieňov

Typy osvetľovaní:

- statické
- dynamické



OSVETĽOVANIE A OSVETĽOVACÍ MODEL

Osvetľovací model je model, ktorým môžeme sledovať vlastnosti povrchu ako je *farba*, *lesklosť*, *matnosť*, *drsnosť* a podobne.

Základom osvetľovacieho modelu je *odrazová funkcia* (reflection function). Je to matematická funkcia vyjadrujúca intenzitu svetelného lúča rozptýleného svetla v závislosti na jeho smere a smere intenzity a vlnovej dĺžky dopadajúceho lúča. Čím lepšie popisuje odrazová funkcia chovanie skutočného svetla, tým lepší a presvedčivejší je dojem z generovanej počítačovej reprezentácie priestorového objektu.

OSVETĽOVANIE

Typy svetelných zdrojov podľa farby:

- Monochromatické
- Achromatické

Typy svetelných zdrojov podľa spôsobu vyžarovania:

- Bodové
- Reflektorové (spot)
- Plošné

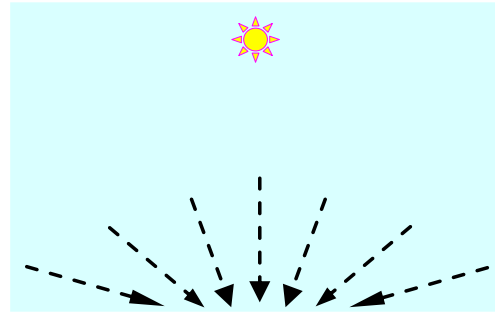
Typy svetelných zdrojov podľa kinematiky:

- Statické
- Dynamické

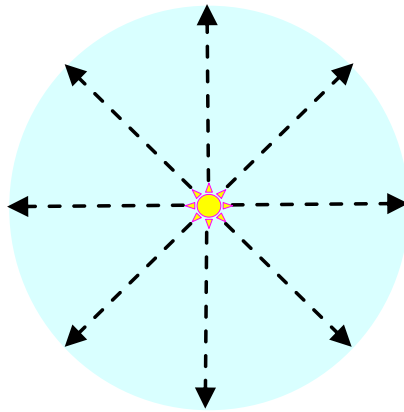


OSVETĽOVANIE – SVETELNÉ ZDROJE

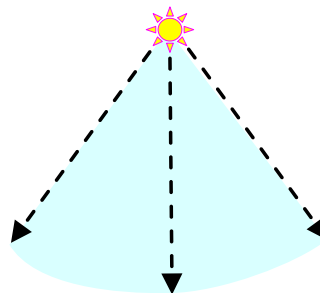
ZÁKLADNÉ ZDROJE



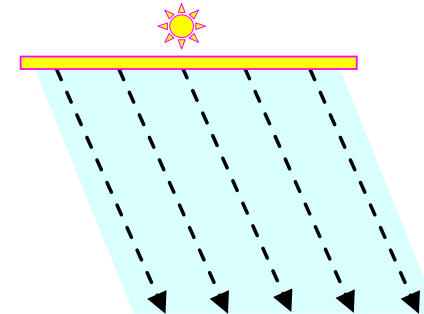
Okolité (ambientné) svetlo
(ambient light)



Bodové svetlo
(point light)



Reflektorové svetlo
(spot light)



Smerové svetlo
(directional light)

OSVETĽOVANIE – ZLOŽKY SVETLA (PHONGOV MODEL)

Ambientná zložka (*ambient light*) sa využíva hlavne pri empirických vyjadrených odrazových rovníc, táto zložka predstavuje odraz bližšie nešpecifikovaného, zo všetkých smerov prichádzajúceho okolitého svetla. Okolité rozptýlené svetlo vzniklo mnohonásobnými odrazmi od ostatných telies a rozptylom spôsobených molekulami vzduchu. Je preto väčšinou biele (achromatické).



OSVETĽOVANIE – ZLOŽKY SVETLA (PHONGOV MODEL)

Difúzna zložka (diffuse) – jej charakteristika nezáleží na smere pohľadu. Po viacnásobnom odraze a lome bude smer svetelného lúča náhodný s rovnakou pravdepodobnosťou všetkých smerov. Veľkosť zložky bude závisieť iba na uhle dopadu (bude úmerná jeho kosínusu). Difúzne odrazené svetlo prinesie informácie o tom, čo nazývame *farba povrchu*.



OSVETĽOVANIE – ZLOŽKY SVETLA (PHONGOV MODEL)

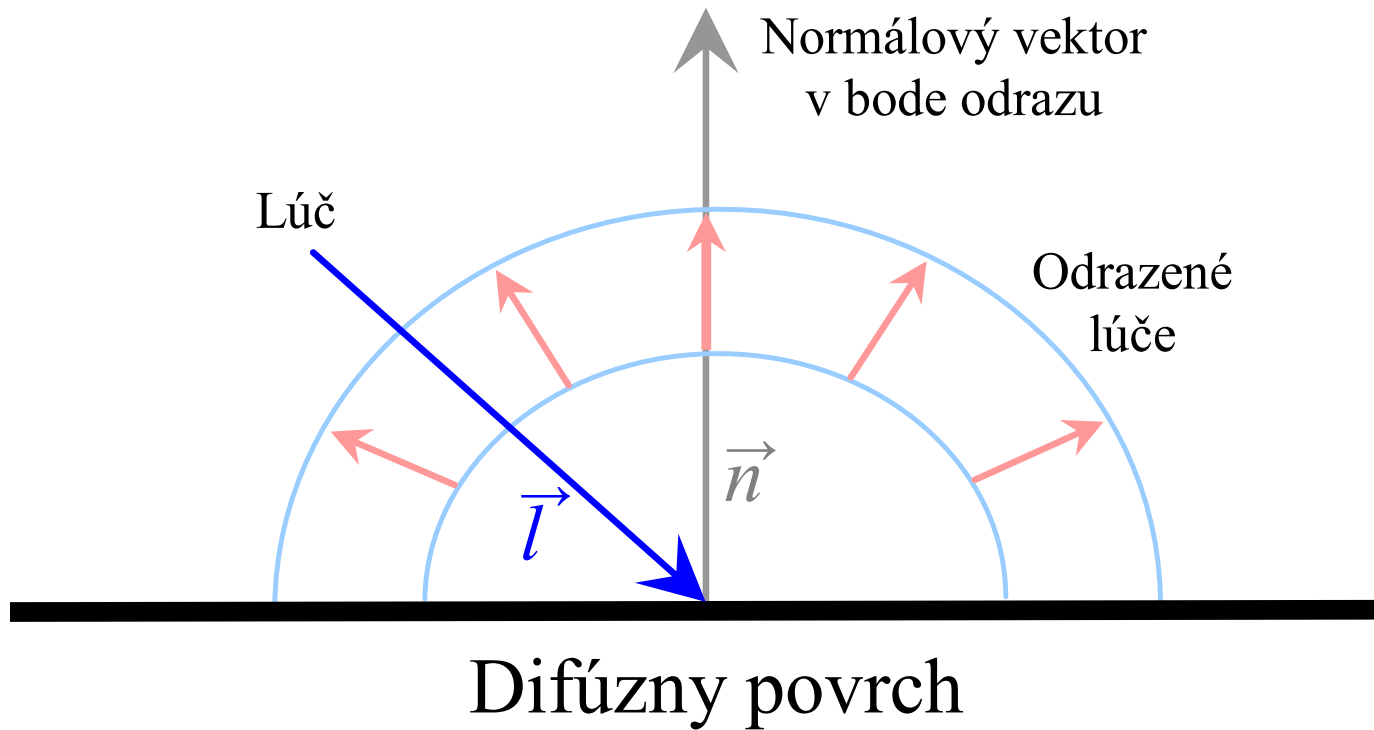
Zrkadlová zložka (*specular*) – jej charakteristika záleží na smere pohľadu. Po viacnásobnom odraze a lome môže byť svetelný lúč utlmený. Pri odraze sa poväčšinou riadi štatistickým rozdelením. Veľkosť zložky bude závisieť na uhle dopadu a optických (zrkadlových) vlastnostiach povrchu.



OSVETĽOVANIE – ZLOŽKY SVETLA (LAMBERTOV MODEL)



J.H. Lambert (1760)



OSVETĽOVANIE – LAMBERTOV OSVETĽOVACÍ MODEL

- Difúzna zložka

$$I_d = L_L r_d (\vec{L} \cdot \vec{N})$$

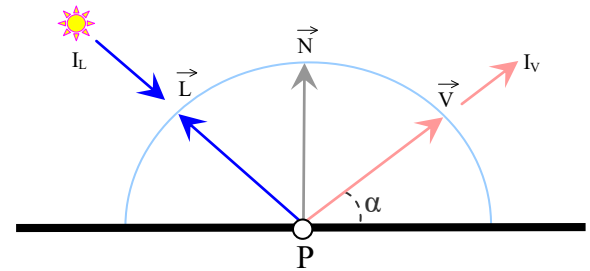
- Ambientná zložka

$$I_a = I_A \cdot r_a$$

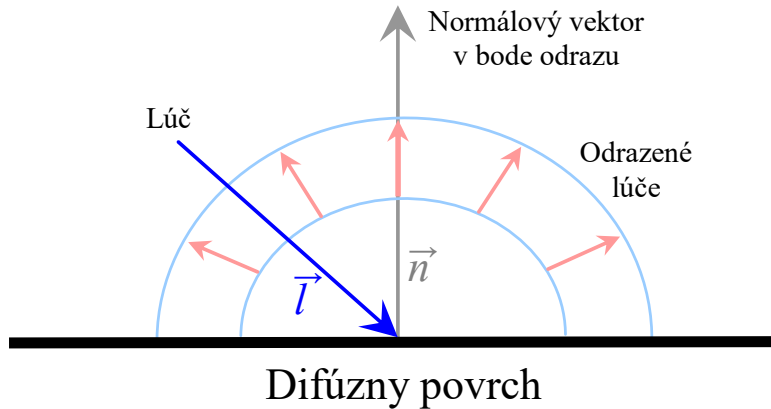
- Výsledok:

$$I_V = I_d + I_a$$

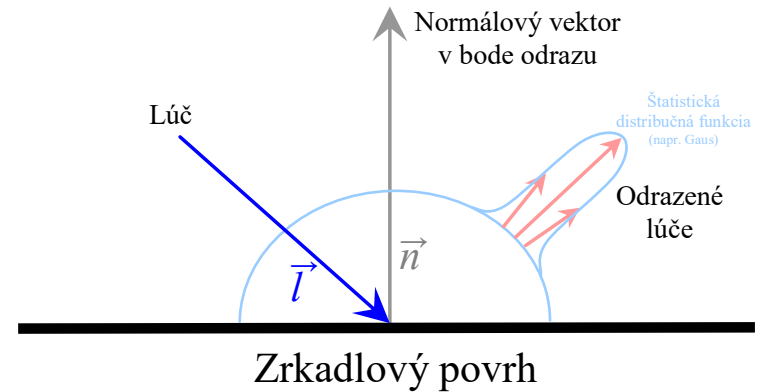
$$I_V = I_A r_a + \sum_{k=1}^M I_{L_k} \left[r_d (\vec{L}_k \cdot \vec{N}) \right]$$



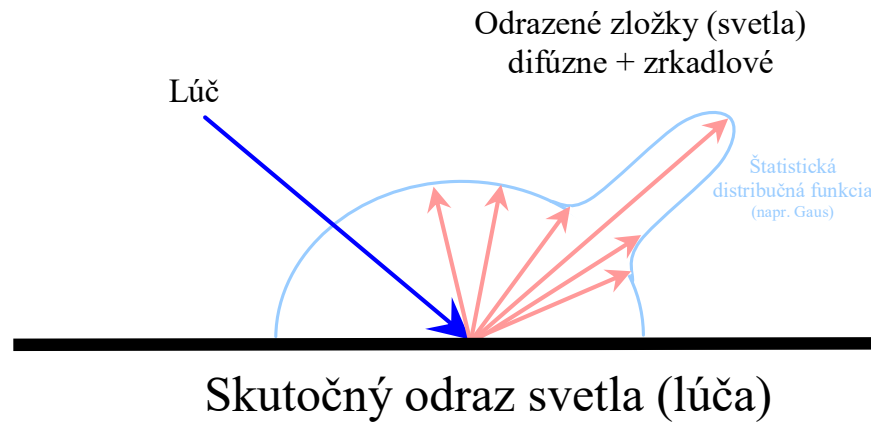
OSVETĽOVANIE – ZLOŽKY SVETLA (PHONGOV MODEL)



+



=



OSVETĽOVANIE – PHONGOV OSVETĽOVACÍ MODEL

- Difúzna zložka $I_d = I_L r_d (\vec{L} \cdot \vec{N})$

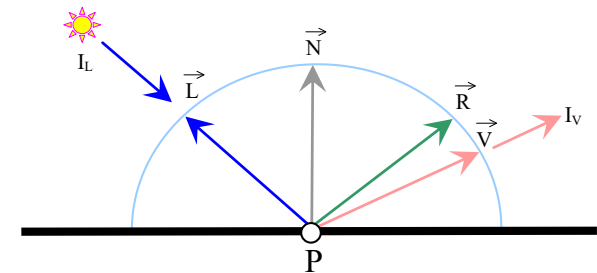
- Zrkadlová zložka $I_s = I_L r_s (\vec{V} \cdot \vec{R})^h$

- Ambientná zložka $I_a = I_A \cdot r_a$

- Výsledok:

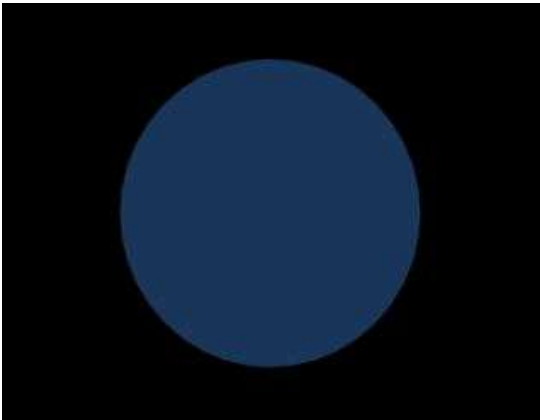
$$I_V = I_s + I_d + I_a$$

$$I_V = I_A r_a + \sum_{k=1}^M I_{L_k} \left[r_s (\vec{V} \cdot \vec{R}_k)^h + r_d (\vec{L}_k \cdot \vec{N}) \right]$$

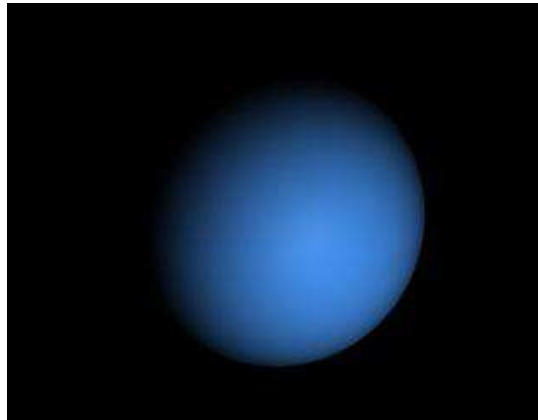


OSVETĽOVANIE – ZLOŽKY SVETLA

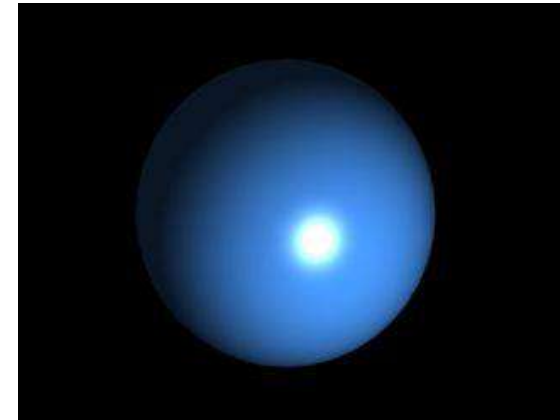
POROVNANIE APLIKÁCIÍ ZLOŽIEK SVETLA



Ambientná
zložka



Ambientna zložka
+
Difúzna zložka



Ambientná zložka
+
Difúzna zložka
+
Zrkadlové zložka

OSVETĽOVACIE MAPY (LIGHTMAPS)

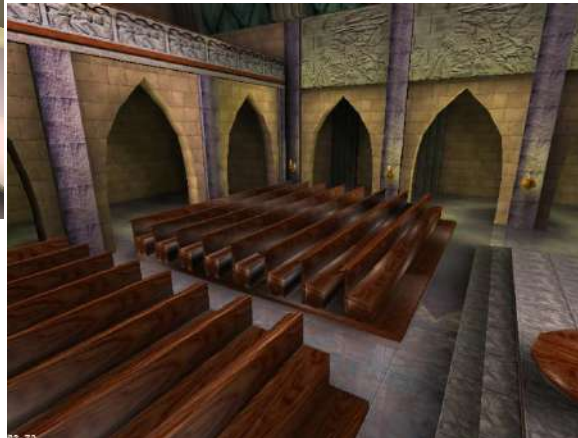
Osvetľovacia mapa je celkový výpočet vplyvov svetelných zdrojov na scénu



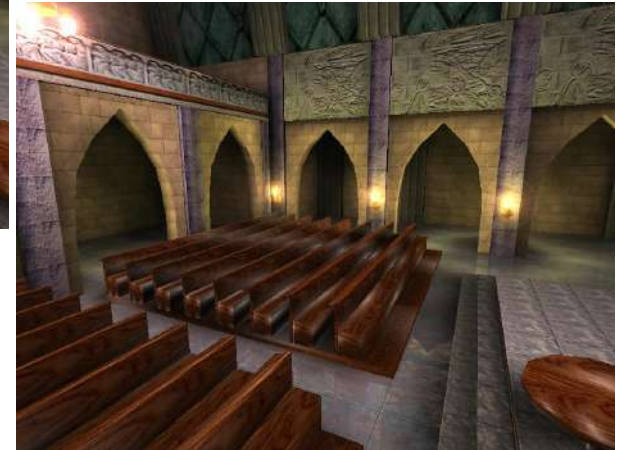
OSVETĽOVANIE - PROCES



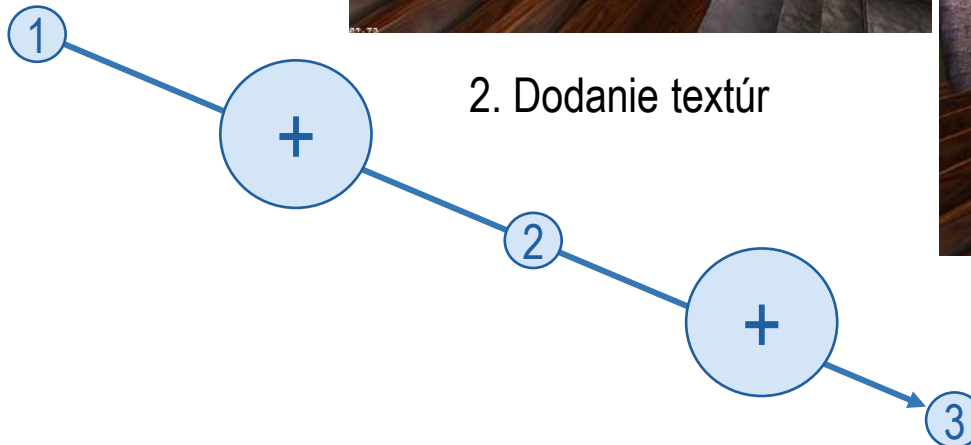
1. Osvetľovacia mapa



2. Dodanie textúr



3. Dodanie svetelných zdrojov



IMPLEMENTÁCIA ZRKADIEL

Odrazy okolitých objektov a svetelných zdrojov na objekte

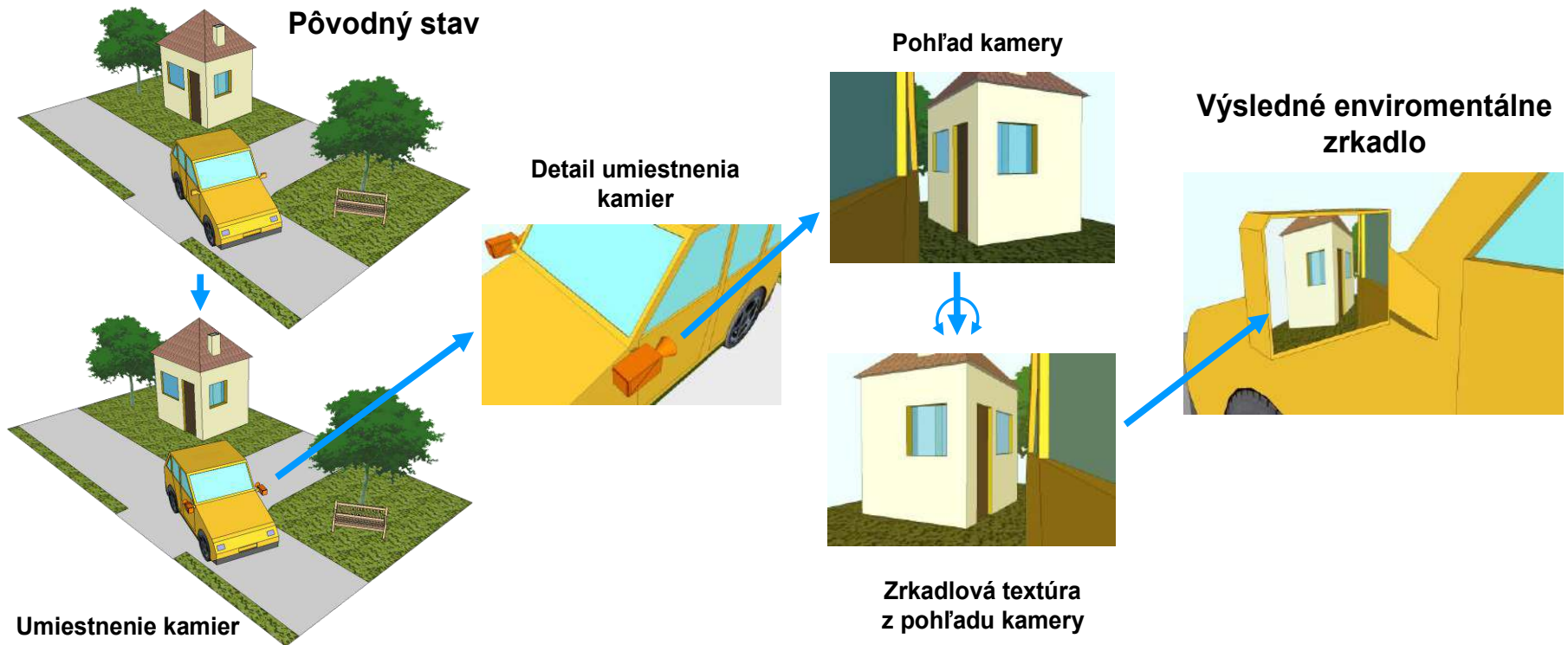
Typy zrkadiel:

- Enviromentálne zrkadlá
- Geometrické zrkadlá



IMPLEMENTÁCIA ZRKADIEL

ENVIROMENTÁLNE ZRKADLÁ (PRÍKLAD)



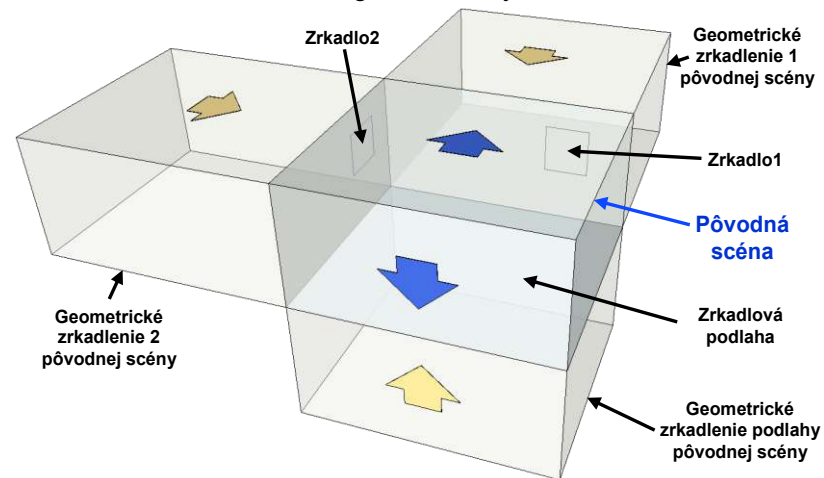
IMPLEMENTÁCIA ZRKADIEL

GEOMETRICKÉ ZRKADLÁ (PRÍKLAD)

Pôvodná scéna



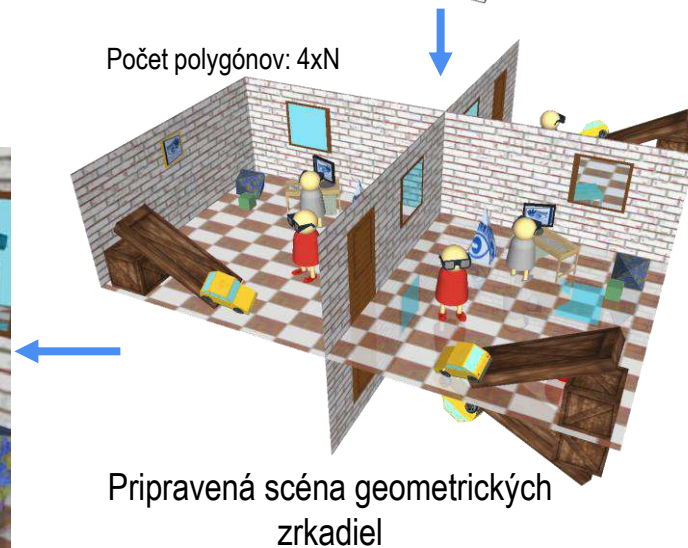
Schéma geometrických zrkadiel



Výsledné geometrické zrkadlá



Počet polygónov: $4 \times N$



GLOBÁLNE OSVETĽOVACIE TECHNIKY (FOTOREALISTIKA)

Globálne osvetľovacie techniky slúžia na riešenie zobrazovacej rovnice (vrátane riešenia viditeľnosti). Ich riešením je:

- Výpočet osvetlenia všetkých plôch v scéne – *pohľadovo nezávislé*.
- Výpočet osvetlenia pre určitý smer – *pohľadovo závislé*.



GLOBALNE OSVETĽOVACIE METÓDY (FOTOREALISTIKA)

Rozdelenie globálnych osvetľovacích techník:

- Metódy vychádzajúce od pozorovateľa
 - Sledovanie lúča
 - Sledovanie cesty
- Metódy vychádzajúce od svetelného zdroja
 - Sledovanie fotónov
 - Monte Carlo sledovanie svetla
- Obojsmerné metódy
 - Obojsmerné sledovanie cesty
 - Fotónové mapy
- Vyžarovacia metóda (Radiosity)



METÓDY VYCHÁDZAJÚCE OD POZOROVATEĽA (FOTOREALISTIKA)

Jedná sa o pohľadovo závislé metódy, ktoré zhromažďujú svetelnú informáciu (energiu), ktorú lúč akumuluje (gathering methods) po svojej trajektórii. Tieto metódy môžeme tiež označiť ako spätné sledovanie trajektórie svetla.

- **Sledovanie lúča** (Ray tracing) - základný algoritmus pracuje s bodovými svetlami, poskytuje len ostré tieňe a transport svetla je obmedzený na zrkadlové odrazy. Je schopný popisovať ľubovoľný počet zrkadlových odrazov a lokálny osvetľovací model s difúznou zložkou. Nepočíta sa kompletne riešenie zobrazovacej rovnice (napr. efekty ako kaustika, plošné svetelné zdroje, difúzne odrazy a pod.).
- **Distribúované sledovania lúča** (Distributed ray-tracing) - ide o rozšírenie Raytracingu o odraz lúča, aj do difúzných smerov, v závislosti od osvetľovacieho modelu.

METÓDY VYCHÁDZAJÚCE OD SVETELNÉHO ZDROJA

(FOTOREALISTIKA)

Tieto metódy sú označované aj ako *energiu vystreľujúce metódy* (*shooting methods*) a riešia problémy predchádzajúcich metód, ako sú kaustika, skryté svetelné zdroje a pod., avšak za cenu zaťaženia šumom. Tieto algoritmy hľadajú riešenie osvetľovacej rovnice náhodným sledovaním dráh lúča zo svetelného zdroja.

- **Photon tracing** - ide o duálnu metódu k metóde sledovania lúča. Používa rovnaké druhy odrazov no v opačnom smere a trajektória sa dá popísať ako $LS \times DE$. Teda nastáva ľubovoľný počet zrkadlových odrazov zo svetla, ktoré ústia do difúzneho odrazu (závislého napr. od hĺbky rekurzie).
- **Light tracing** - Metóda sledovanie svetla je duálna k metóde sledovania cesty (path tracing). Náhodným spôsobom sa určujú odrazy lúčov v scéne, ktoré začínajú zo svetelného zdroja. Pokiaľ je bod odrazu lúča viditeľný pozorovateľom, určí sa jeho príspevok pre daný pixel (všetky príspevky pre daný pixel sa spriemerujú). Trajektória sa ukončí, pokiaľ bude príspevok svetla dostatočne malý.

ALGORITMY FOTOREALISTICKÝCH METÓD

Algoritmy zobrazujúce povrchy (surface based techniques)

Algoritmy zobrazujúce povrchy vytvárajú pomocnú geometrickú reprezentáciu povrchu. V trojrozmerných dátach sa hľadajú hrany a body povrchu. Z nich sa interpretuje povrch dvojrozmernými záplatami. Ide o nepriamu metódu. Medzi najznámejšie algoritmy patria:

- sledovania obrysov (*contour tracking*)
- marching cubes (*pochodujúce kocky*)
- marching tetrahedra
- dividing cubes
- opaque cubes

Tieto metódy sa snažia z objemových dát aproximovať povrch pomocou geometrických primitív a tie zobrazovať pomocou dobre známych metód počítačovej grafiky. Informácie o vnútrajšku objektu sa však stratia.

ALGORITMY FOTOREALISTICKÝCH METÓD

Objemové algoritmy

Využívajú plnú priestorovú informáciu na vykreslenie výsledného obrazu a sú nezávislé od zložitosti scény. Na výpočet používajú celú trojrozmernú mriežku údajov, preto sú veľmi náročné na pamäť a procesor. Pri každom výslednom obraze je vyžadované *traverzovanie údajov*. Je možné zobrazit' ľubovoľný detail objektu. Tieto algoritmy poskytujú viac informácií ako algoritmy, ktoré používajú pomocnú reprezentáciu povrchu. Podľa klasifikácie údajov sa delia na:

- **Binárne** - pokrývajú každý voxel buď úplne alebo vôbec. Sú to povrchovo orientované algoritmy
- **Pravdepodobnostné** - priradujú voxelom percentuálny podiel nejakého objektu. Sú založené na nahradzovaní príspevkov od všetkých vzoriek pozdĺž lúča do jedného obrazového pixela.

Ďalšie rozdelenie je podľa poradia, v akom sú voxely postupne spracované pri tvorbe obrazu, čiže ide o delenie podľa domény algoritmu.

ALGORITMY FOTOREALISTICKÝCH METÓD

Algoritmy pracujúce v obrazovom priestore

Pre každý pixel výsledného obrazu sa hľadajú voxely v objektovom priestore, ktoré prispievajú do výslednej farby bodu. Nakoľko pozície týchto voxelov obyčajne padnú mimo vrcholu mriežky, ich hodnotu musíme zisťovať interpoláciou. Do tejto skupiny môžeme zaradiť nasledujúce algoritmy:

- Trasovanie lúčov (raytracing, raycasting)
- Sábellova metóda

ALGORITMY FOTOREALISTICKÝCH METÓD

Algoritmy pracujúce v objektovom priestore

Pre každý voxel objektového priestoru sa hľadajú pixely výsledného obrazu, ktoré daný voxel ovplyvnia. Splatting je technika traverzujúca objektový priestor. Na každý voxel je aplikovaná konvolúcia s 3D rekonštrukčným filtrom. Príspevok filtrovaných bodov sa nahromaďuje do obrazového priestoru.

Algoritmy:

- V-buffer
- Splatting

ALGORITMY FOTOREALISTICKÝCH METÓD

Algoritmy pracujúce na hybridnom princípe

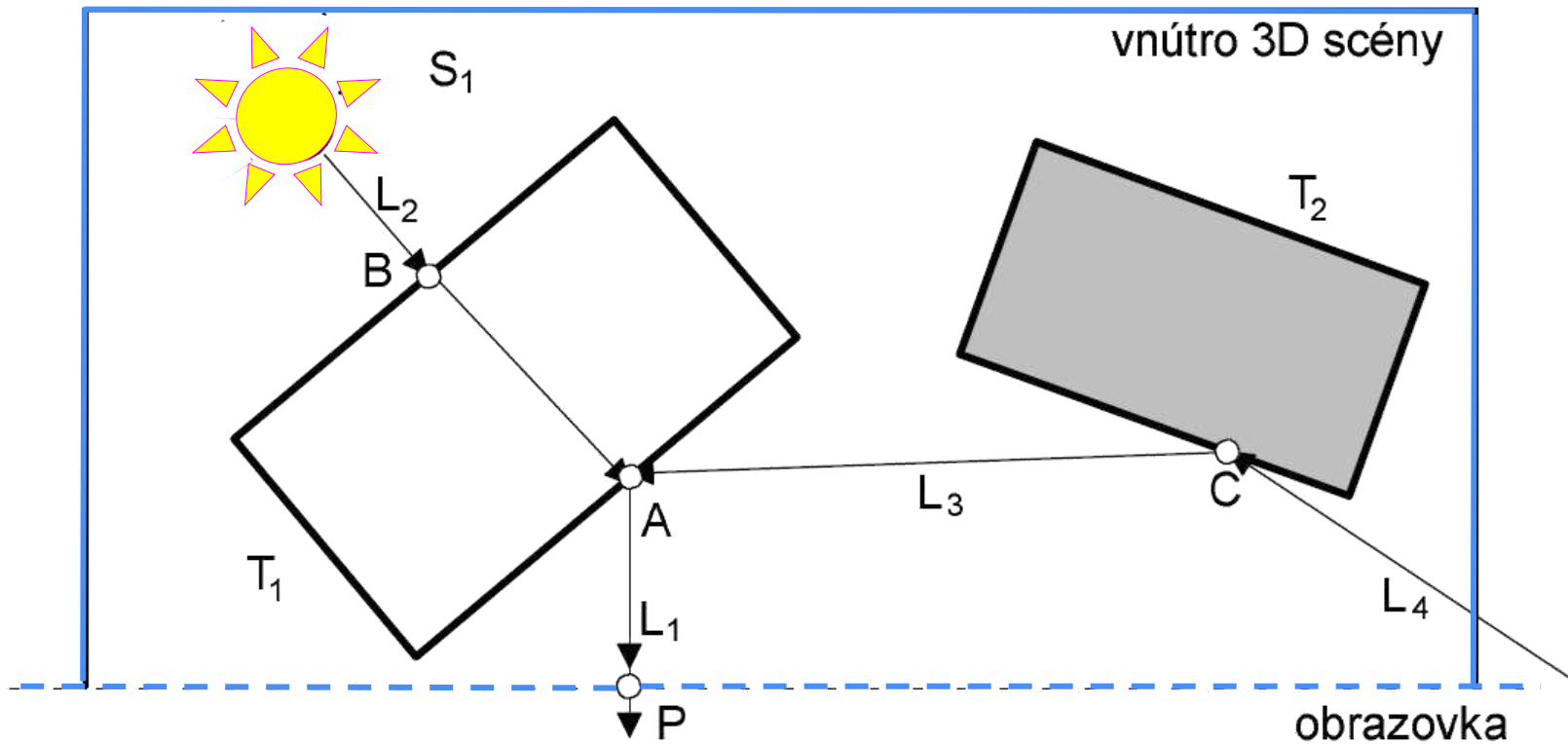
Kombinujú výhody oboch predchádzajúcich prístupov. Napríklad sa môže použiť algoritmus *Shear-warp* (posuň a pokrív), ktorý je považovaný za najrýchlejší. Myšlienka spočíva v posunutí jednotlivých rezov v objektovom priestore tak, aby ich namapovanie na 2D zobrazovací priestor bolo jednoduché a rýchle – vzorky pozdĺž lúčov budú ležať presne v rovinách rezov.

Traverzovanie údajmi vyžaduje jednoduché adresovanie a 2D prevzorkovací filter. Po projekcii sa zdeformovaný obraz pokríví naspäť. Algoritmus je veľmi rýchly aj na bežne dostupných počítačoch. Na druhej strane však produkuje artefakty, je pohľadovo-závislý a kvalita výsledného obrazu je nízka. Nižšia kvalita obrazu je spôsobená tým, že:

- algoritmus obsahuje až dva kroky prevzorkovania, čo môže spôsobiť rozmazanie a stratu detailných informácií,
- rekonštrukčný filter je iba dvojrozmerný. V rezoch sa používa iba bilineárna interpolácia, medzi rezmi iba interpolácia najbližším susedom. Tento bod je hlavnou nevýhodou algoritmu *shear-warp*,
- počet lúčov je rovný počtu voxelov jedného rezu, algoritmus produkuje alias kvôli podvzorkovaniu.

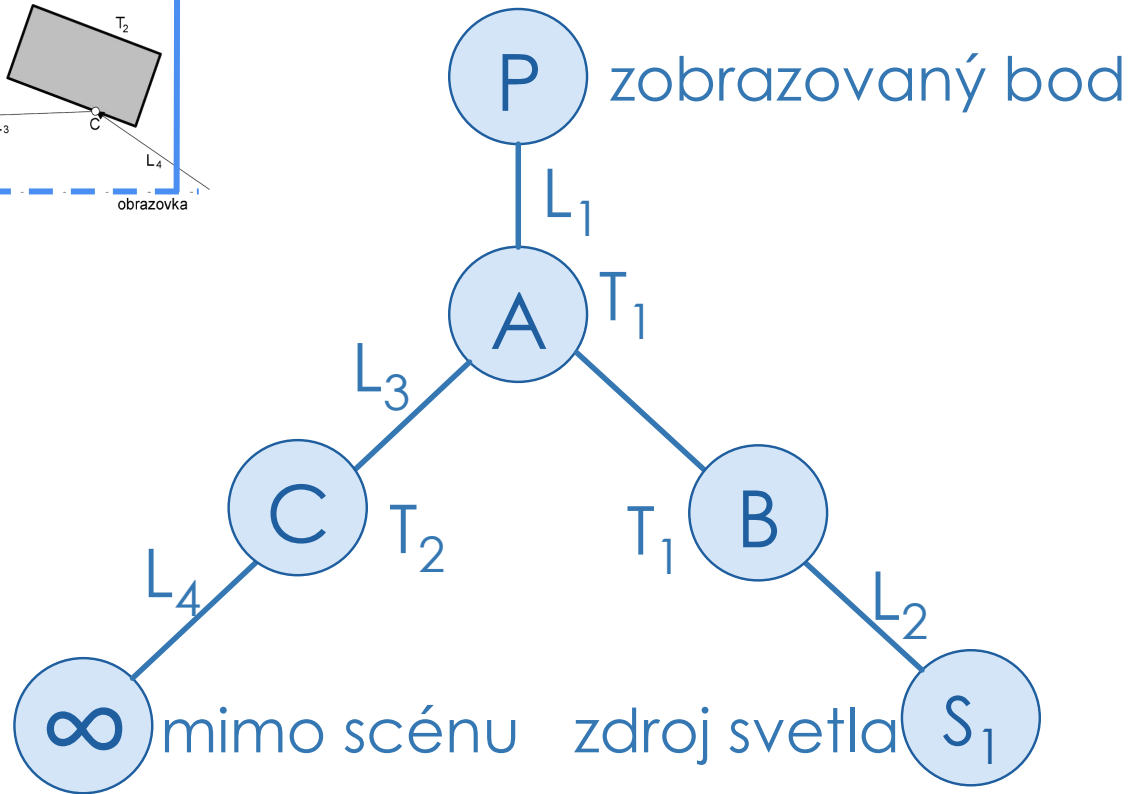
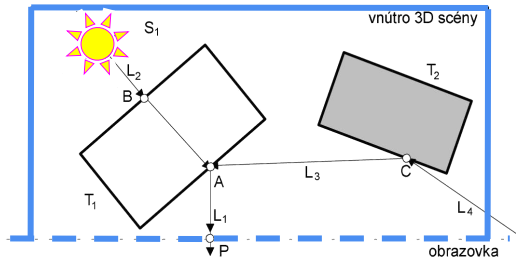
RAYTRACING

SPÔSOB SLEDOVANIA LÚČA PRI RAYTRACING-U



RAYTRACING

PRÍKLAD STROMU SCÉNY PRI RAYTRACING-U



RAYTRACING

PRÍKLAD VÝPOČTU INTENZITY PRI RAYTRACING-U

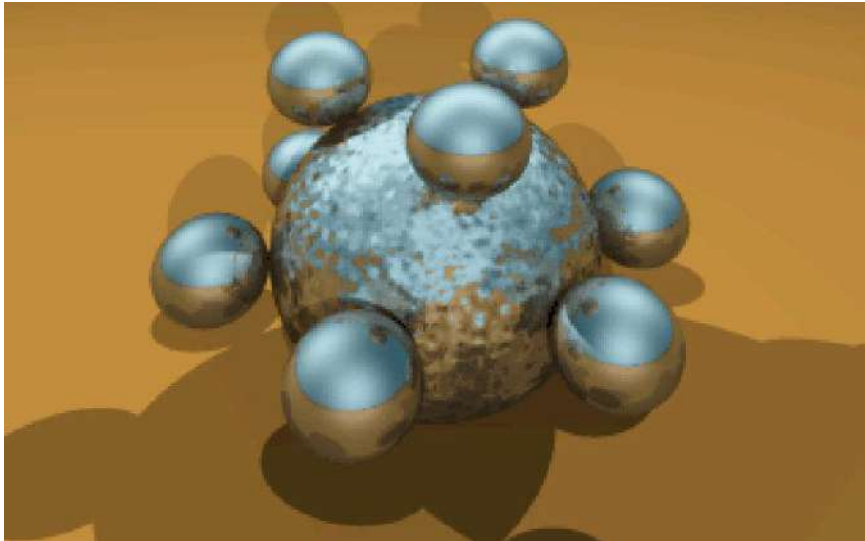
$$I_V = I_a + I_d + I_s + I_r + I_L$$

kde:

- I_V je výsledná intenzita
-
- I_a je zložka okolitého svetla
 - I_d je difúzna zložka
 - I_s je zrkadlová zložka (týka sa zdroja svetla)
 - I_r je zložka z odrazeného lúča (týka sa telesa)
 - I_L je zložka z lomeného lúča

PRÍKLADY SCÉN (FOTOREALISTIKA)

PRÍKLAD VÝSTUPOV Z RAYTRACINGU A VYŽAROVACEJ METÓDY



PRÍKLADY SCÉN (FOTOREALISTIKA)

PRÍKLAD VÝSTUPOV Z ROZSIAHLEHO RAYTRACINGU



Priestory KPI (30Kx20K)



Tepláreň Košice (TEKO)



Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025

APLIKÁCIE POČÍTAČOVEJ GRAFIKY A VIRTUÁLNA REALITA

doc. Ing. Branislav Sobota, PhD.

Katedra počítačov a informatiky, FEI TU v Košiciach

P 10

© 2025

POUŽITIE POČÍTAČOVEJ GRAFIKY

- Forma komunikácie s počítačom a používateľské rozhranie
- Obaly v obchodoch
- Design automobilov či výrobkov spotrebnej elektroniky
- Projektovanie budov a interiérov
- Noviny, časopisy, katalógy
- Predpoved' počasia
- Počítačové hry a zábava
- Spracovanie hospodárskych či štatistických výsledkov
- Film, video, reklama
- ...



TRENDY POUŽITIA POČÍTAČOVEJ GRAFIKY

- počítačové hry
- fotorealistické zobrazovanie, multimédiá, virtuálna realita a používateľské rozhrania
- manažérska grafika a reklama
- počítačom podporovaný návrh a konštruovanie
- vizuálna simulácia
- spracovanie obrazu



Počítačové hry

sú samostatnou kapitolou. Je možné povedať, že vlastne sprostredkovávajú prvý kontakt neznalejšieho človeka s počítačom. Počítačové hry majú svoj základ už na prvých sálových počítačoch. Ich rozmach ale nastal až po nástupe malých domácich (8-bitových) počítačov. Základné počítačové hry dosiahli svoj kulminačný bod práve však na osobných počítačoch. Stále viac a viac sa presadzuje čo dokonalejšia grafika a je jasný trend prechodu od klasických dvojrozmerných hier ku trojrozmerným priestorovým hrám vrátane mobilných platforiem.



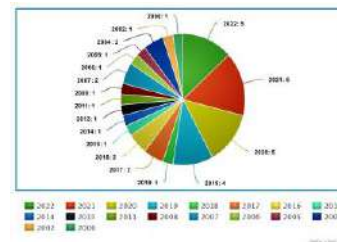
FOTOREALISTIKA, MULTIMÉDIÁ, VIRTUÁLNA REALITA

sú hitom dnešnej doby. U prvého ide o čo najvernejšie zobrazenie priestorových scén a objektov vrátane osvetlenia a riešenia viditeľnosti. Dokonalé skĺbenie kvalitného obrazu a zvuku je doménou multimédií. Ak sa ku kvalitnému programu pridá dostatočne výkonný počítač, ktorý umožňuje toto prepočítavanie v reálnom čase je možné simulovať v danom čase neexistujúci svet (virtual reality), napr. "prechádzať sa" ešte v nepostavenom vlastnom dome Informačné systémy a počítačové siete v spolupráci s prostriedkami virtuálnej reality (najmä formu netradičného rozhrania komunikácie človek počítač) začínajú tvoriť nový fenomén veľmi blízkej budúcnosti.



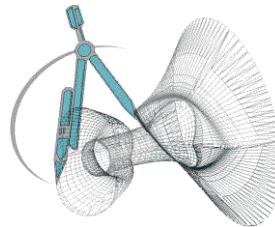
MANAŽÉRSKA GRAFIKA A REKLAMA

slúži na grafické znázornenie výsledkov napr. rôznych kalkulácií alebo predstavenie produktu. Používateľ má obvykle možnosť výberu z niekoľkých typov diagramov napr. kruhové, stĺpcové a pod., alebo možnosť tzv. ikonickej grafiky (symbolické zobrazenia napr. postava, strom a.i. Potom napr. jedna postava môže znázorňovať určitý počet ľudí/km²). Príslušné merítka a transformácie väčšinou vypočítava systém sám. Predstavenie produktu využíva väčšinou hybridný prístup k vizualizácii.



POČÍTAČOM PODPOROVANÝ NÁVRH

(CAD - Computer Aided Design) našlo v počítačoch veľmi dobrú "pôdu". Od jednoduchých dvojrozmerných grafických editorov cez trojrozmerné až ku veľkým návrhovým komplexom najmä z hľadiska lepšej konštruktárskej predstavivosti alebo reálnejšiemu pohľadu. V súčasnosti je asi už pokrytá každá oblasť výroby príslušným CAD-ovským programom. Svoje prvé uplatnenie však mali tieto programy najmä v oblasti elektrotechniky, strojárstva a stavebníctva.



VIZUÁLNA SIMULÁCIA

je oblasť počítačovej grafiky, ktorá sa rozvíja najmä v dvoch smeroch. Prvým je vizualizácia výsledkov rôznych simulačných programov napr. priebehy napätí a prúdov v elektrických obvodoch. Druhým je simulácia reálnych udalostí v rôznych počítačových trenažéroch napr. leteckých alebo automobilových.



SPRACOVANIE OBRAZU

je časťou počítačovej grafiky, do ktorej zasahujú aj iné odbory. V princípe sem môžeme zahrnúť najmä procesy spracovania farieb, detekcie hrán, vyhladzovanie obrazu a pod. Vo veľkej miere sa používajú tieto postupy napr. v grafických editoroch ale aj v náročných vojenských alebo vesmírnych projektoch, napr. na vylepšenie prichádzajúcich snímok alebo v procese rozpoznávania obsahu obrazu a umelej inteligencie.

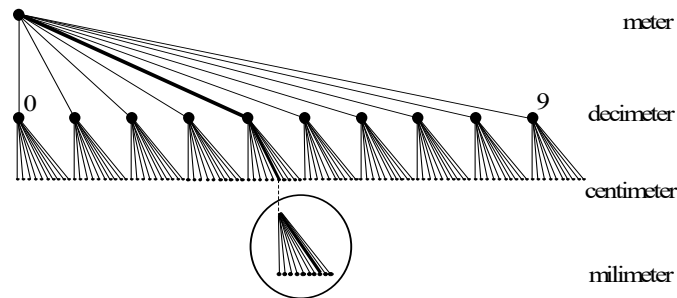


FRAKTÁLY

„Fraktál je množina, ktorej Hausdorffova dimenzia je väčšia než dimenzia topologická.“

(Mandelbrot)

Fraktál je odvodený od latinského slova **frangere**, ktoré znamená zlomiť (angl. to break).

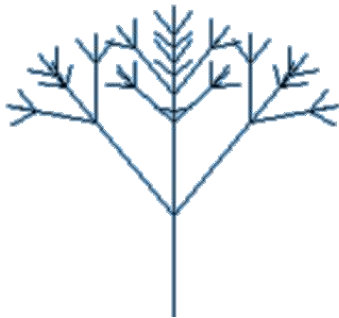


Sebepodobnosť

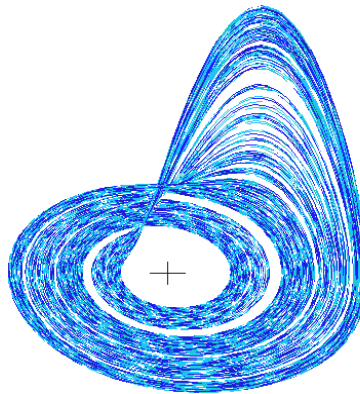
Jeden zo základných pojmov v oblasti fraktálov a fraktálnej geometrie je *sebepodobnosť*. Môžeme ju nájsť na rôznych miestach okolo nás. Zoberme si hlávku karfiolu. Po podrobnejšom preskúmaní je možné si všimnúť, že sa skladá z menších častí, ktoré tvoria celú hlávku karfiolu. Oddelíme jednu z týchto častí tvoriacich prvotnú hlávku. Zistíme, že je veľmi podobná pôvodnému celku, ktorý tvorila celá hlávka karfiolu. Jediný rozdiel je iba vo veľkosti. Tento postup je možné opakovať niekoľkokrát za sebou. Po každom takomto kroku je možné sledovať vzájomnú podobnosť oddeľovaných častí, pričom bude vždy platiť, že sa odlišujú iba vo veľkosti.

FRAKTÁLY - TYPY

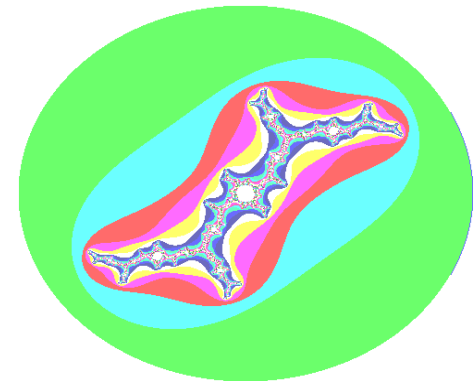
1. L-systémy (Linden-mayerove systémy) definované cez pravidlá prepisu
2. IFS (systém iterovaných funkcií (Iterated Function Systems)) definované iterovanou aplikáciou afinných transformácií
3. Dynamické systémy (dynamické množiny (Dynamic sets)) definované iteráciou komplexných funkcií



1.



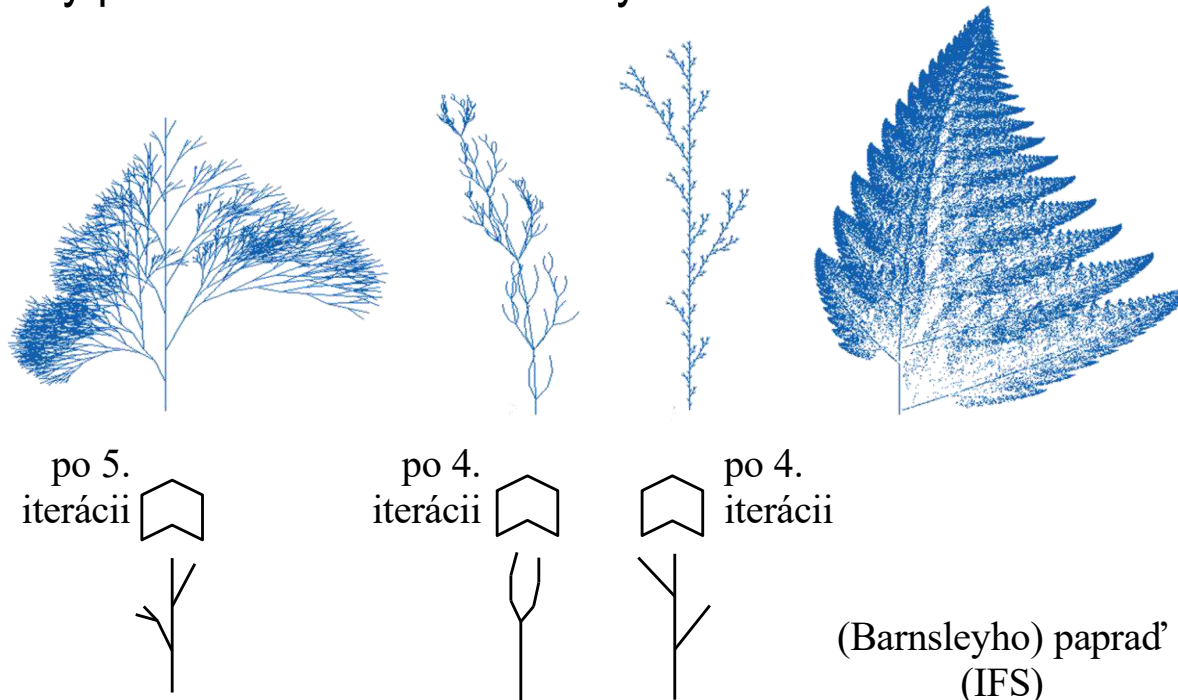
2.



3.

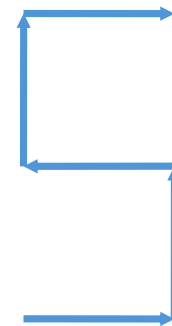
L-SYSTÉMY

- Linden-mayerove systémy a patria medzi najjednoduchšie fraktály.
- Sú vhodné na popis objektov prírody.
- Je to systém aplikujúci pravidlá nezávisle na poradí.
- L-systém je definovaný svojím stavom a tabuľkou akcií.
- Stav je popísaný polohou a orientáciou L-systému.

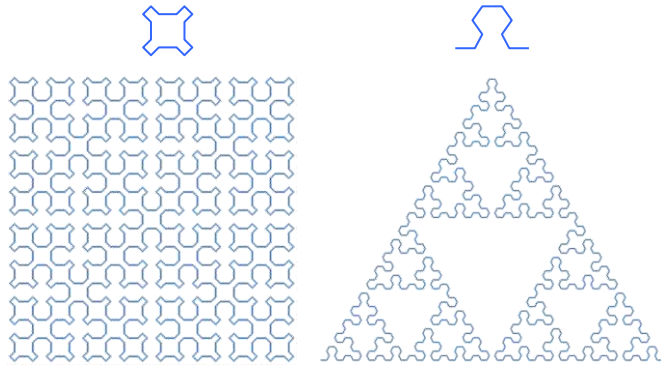


L-SYSTÉMY

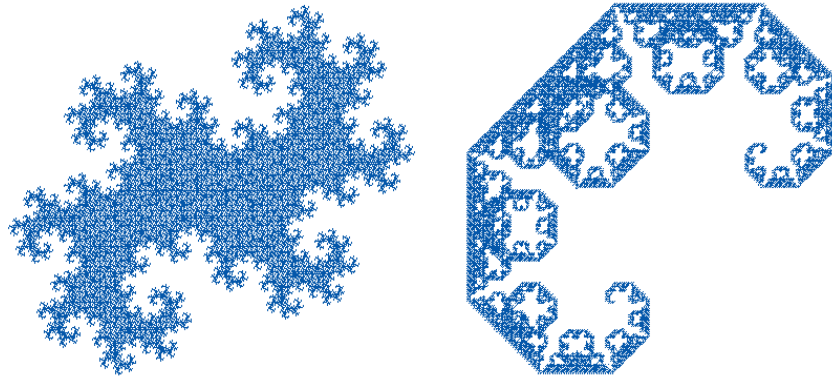
- Grafickou interpretáciou L-systémov je najčastejšie tzv. *korytnačia grafika*.
- Je založená na popise fraktálov pomocou programovacieho *jazyka LOGO*, ktorý umožňoval programovanie na základe ovládania pohybu korytnačky.
- Celý tento popis bol prevzatý a použitý pre popis generovania fraktálov.
- Na ovládanie pohybu boli zavedené *štyri základné pravidlá*:
 - presunutie dopredu (*F-Forward*) o presne stanovený krok dĺžky l , pričom sa vykresľuje čiara.
 - presunutie vpred (f) ako pri F , ale čiara sa nekreslí.
 - otočenie vľavo (+) o dopredu stanovený uhol δ .
 - otočenie vpravo (-) o dopredu stanovený uhol δ .



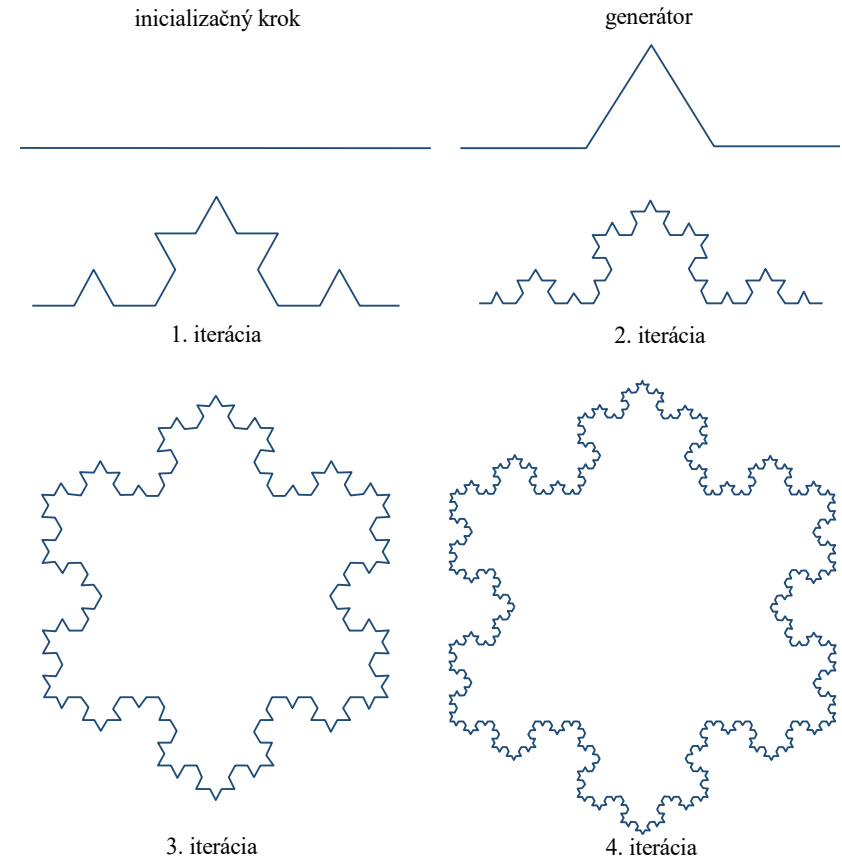
L-SYSTÉMY - PRÍKLADY



Sierpinského vzory



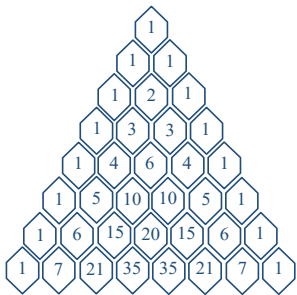
Dračia a C krivka



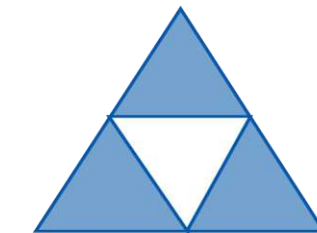
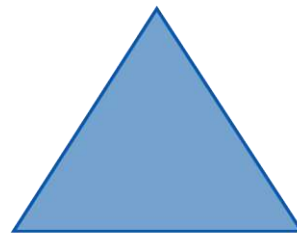
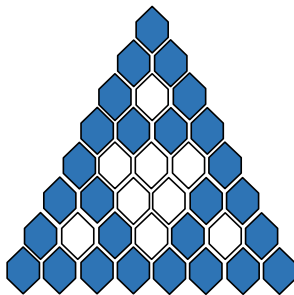
Kochova krivka a vločka

IFS (ITERATED FUNCTION SYSTEMS)

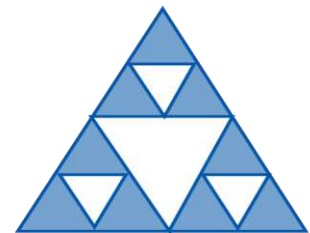
- *Generatívny typ fraktálov, pričom doslovný preklad by bol systém iterovaných funkcií.*
- Tento typ fraktálov má veľké použitie napr. pri *fraktálnej komprimácii údajov.*
- V tomto prípade sa na obrazovke *iteratívnym spôsobom kreslia body (útvary).*



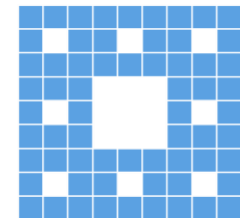
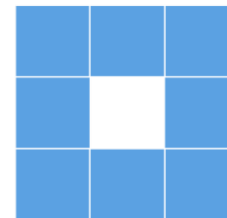
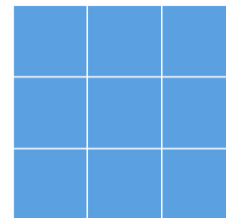
Pascalov trojuholník (kombinatorická štruktúra vytvárajúca IFS fraktálne vzory)



Sierpinského trojuholník



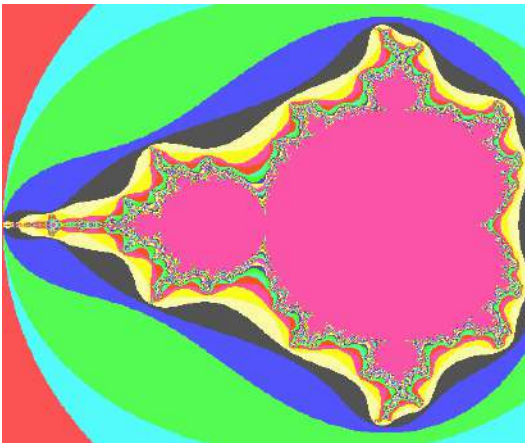
Mengeho huba



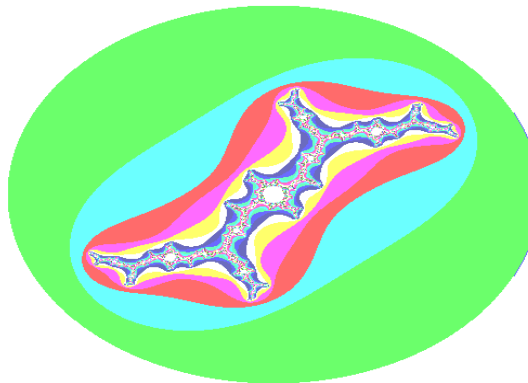
Sierpinského koberček

DYNAMICKÉ MNOŽINY

- *Vzhľadovo najzaujímavejšie typy fraktálov, ktoré sú najčastejšie rovinným obrazom trojrozmerných objektov.*
- *Tretí rozmer sa znázorňuje farebne, pričom farba predstavuje výšku v danom bode.*
- *Zakladajú sa na poznaní komplexných čísel, ktoré sú vlastne iba rozšírením množiny reálnych čísel.*



Mandelbrotova množina



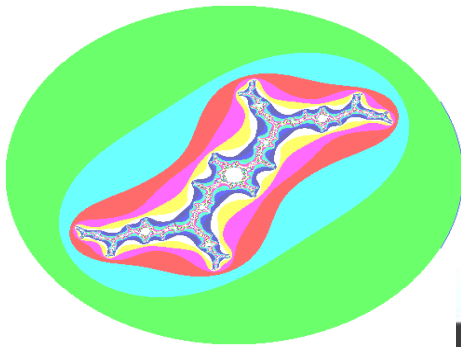
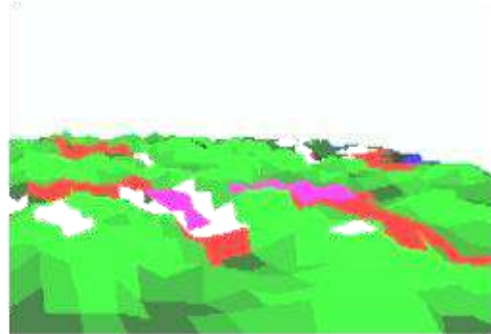
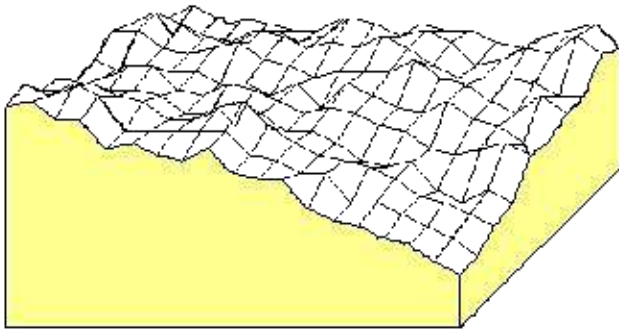
Juliova množina



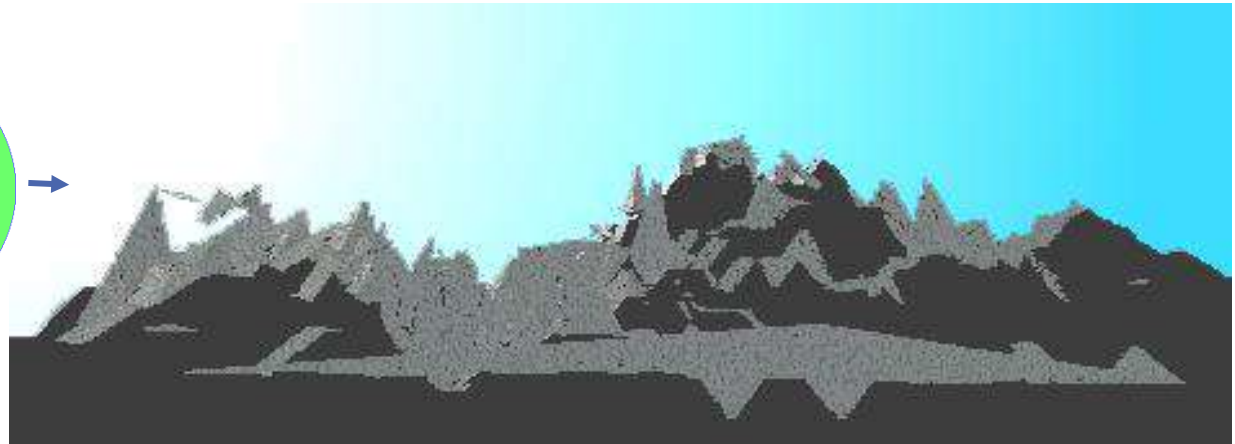
Plazma

FRAKTÁLY A PRÍRODNÉ ÚTVARY A JAVY

KRAJINA

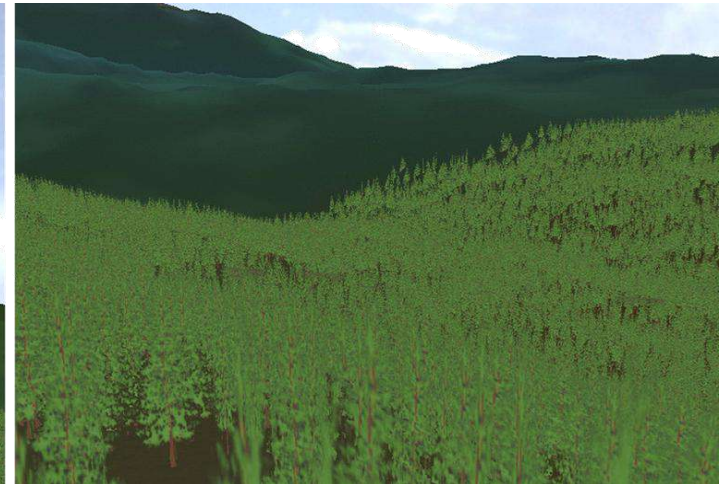
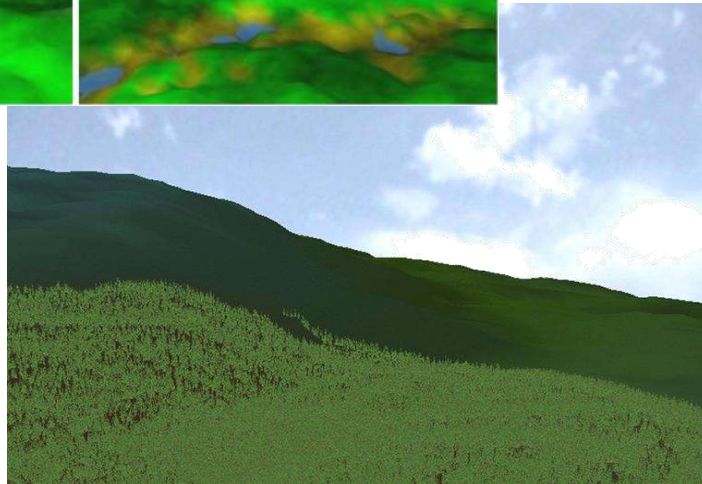
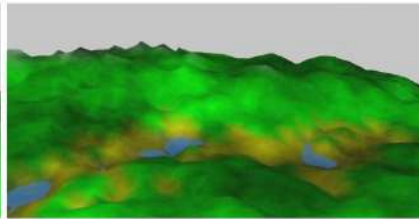
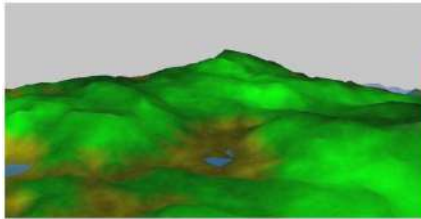
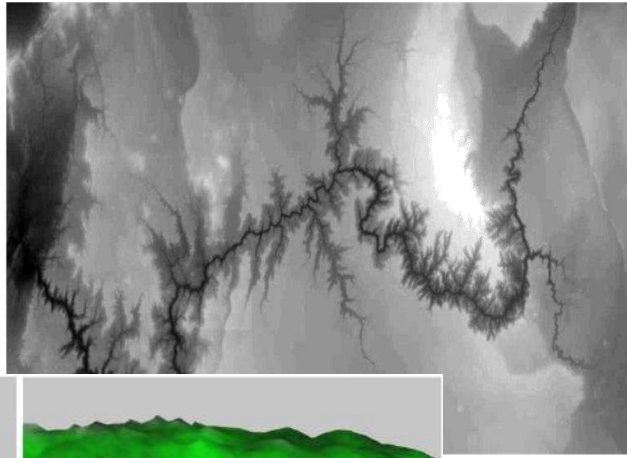


Juliova množina



FRAKTÁLY A PRÍRODNÉ ÚTVARY A JAVY

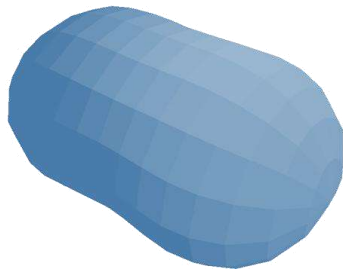
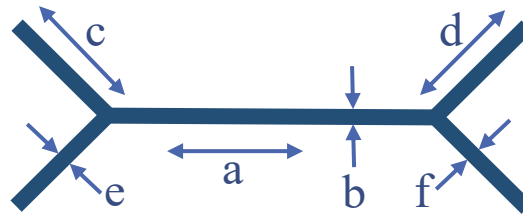
PLOCHY A PRÍRODNÉ ÚTVARY



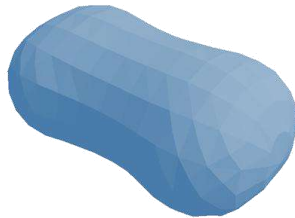
Ukážka vizualizácie povrchu (NURBS povrch) fraktálne generovanej krajiny

FRAKTÁLY A PRÍRODNÉ ÚTVARY A JAVY

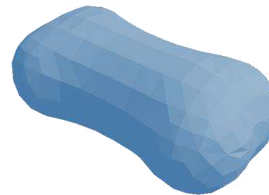
KOSTI



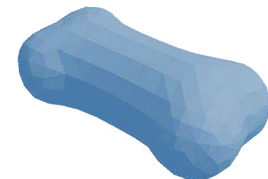
1



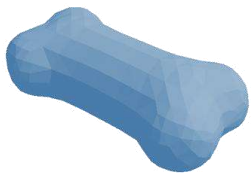
2



3



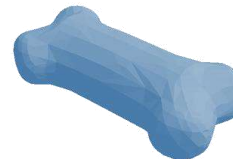
4



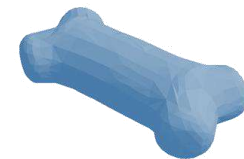
5



6



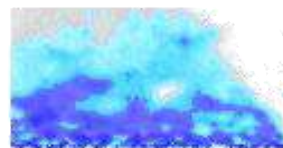
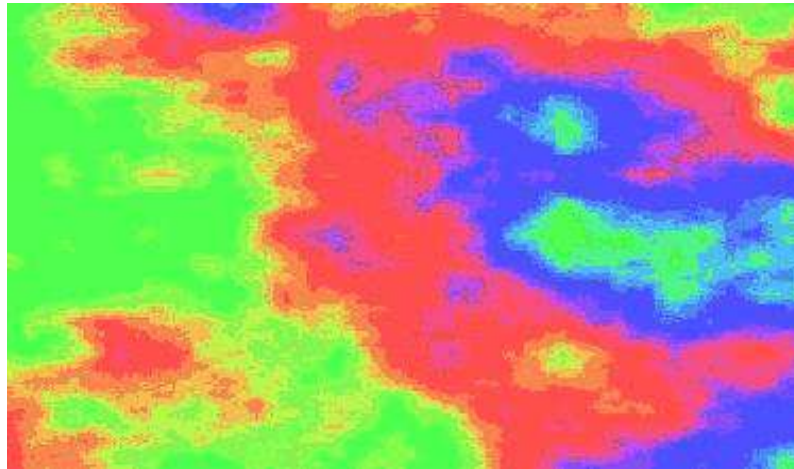
7



8

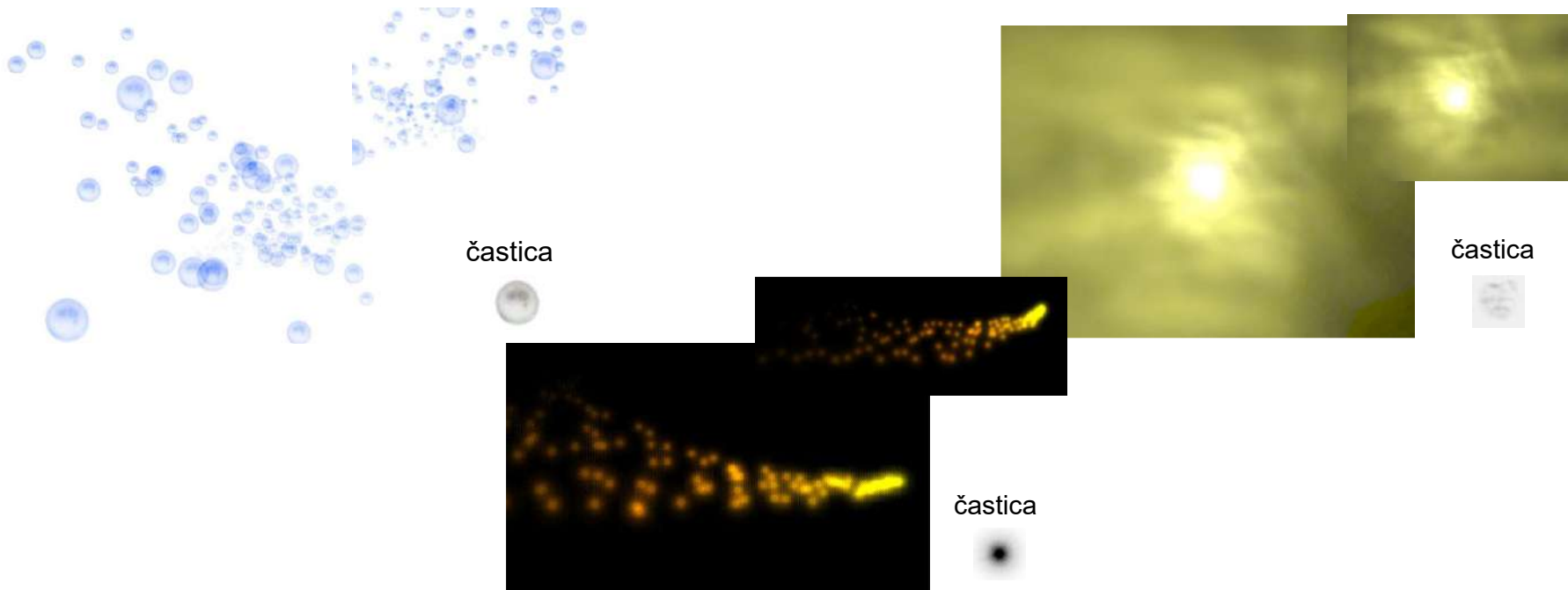
FRAKTÁLY A PRÍRODNÉ ÚTVARY A JAVY

PLAZMA (AMORFNÉ TVARY, OHEŇ, OBLOHA)



ČASTICOVÉ SYSTÉMY (PARTICLE SYSTEMS)

Častica je objekt v scéne, ktorý má svoj vlastný životný cyklus. Niekde sa „narodí“, počas celého svojho života sa vyvíja podľa prideleného kódu a na konci „umiera“. To, čo vykonáva alebo ako sa vyvíja častica je plne v réžii programátora. Časticové systémy (particle systems) sa pre niektoré prírodné javy používa častejšie ako fraktály.



VIRTUÁLNA REALITA

Virtuálno-realityný systém predstavuje interaktívny počítačový systém, vytvárajúci ilúziu v danom čase neexistujúceho len syntetizovaného priestoru alebo ešte presnejšie môžeme hovoriť o tzv. dokonalej simulácii v prostredí tesného spojenia človek-výpočtový systém.

Základné atribúty VR:

- **Imerzia** – pohltenie používateľa resp. jeho pocit pohltenia, vnímanie prostredia emocionálne
- **Percepcia** – prítomnosť používateľa v prostredí, vnímanie prostredia dimensionálne a
- **Interakcia** – komunikácia s prostredím



POUŽÍVANÉ POJMY

- Virtual reality
 - Fuzzy reality
 - Zmiešaná realita (mixed reality, MR)
 - Virtualizovaná realita (virtualised reality, VdR, iba na báze reality)
 - Obohatená realita (enriched reality, ER)
 - Rozšírená realita (augmented reality, AR)
 - Rozšírená virtualita (augmented virtuality, AV)
 - Zdieľaná (kolaboratívna) realita (SVR/CVR)
 - eXtended Reality – Cross-Reality (XR, Virtual+Augmented+Mixed=eXtended) zjednotená aplikačná platforma
 - Hyper reality (kombinácia VR, MR, AR, technolgicky, vrátane fyzickej kombinácie)
 - Metaverzum (Metaverse)
-
- Cyberspace
 - Pozorovateľ, cybernaut
 - Avatar (zástupca pozorovateľa vo virtuálnom prostredí)
-
- HCI – Human Computer Interaction
Interakcia človeka s počítačom sa zaoberá návrhom, vyhodnotením a realizovaním interaktívnych výpočtových systémov pre použitie človekom a štúdiom hlavných javov, ktoré ich obklopujú.



AVATAR

Psycho-morfologická štruktúra reprezentujúca používateľa vo virtuálnom svete

Základné vlastnosti avatara:

- Morfologicko-vizuálne
- Schopnosti/možnosti vo virtuálnom svete
- Limity vo virtuálnom svete
- Implementačná štruktúra

Niekedy sa technológia avatara kombinuje s technológiou *digitálneho dvojčaťa (Digital twin)*.



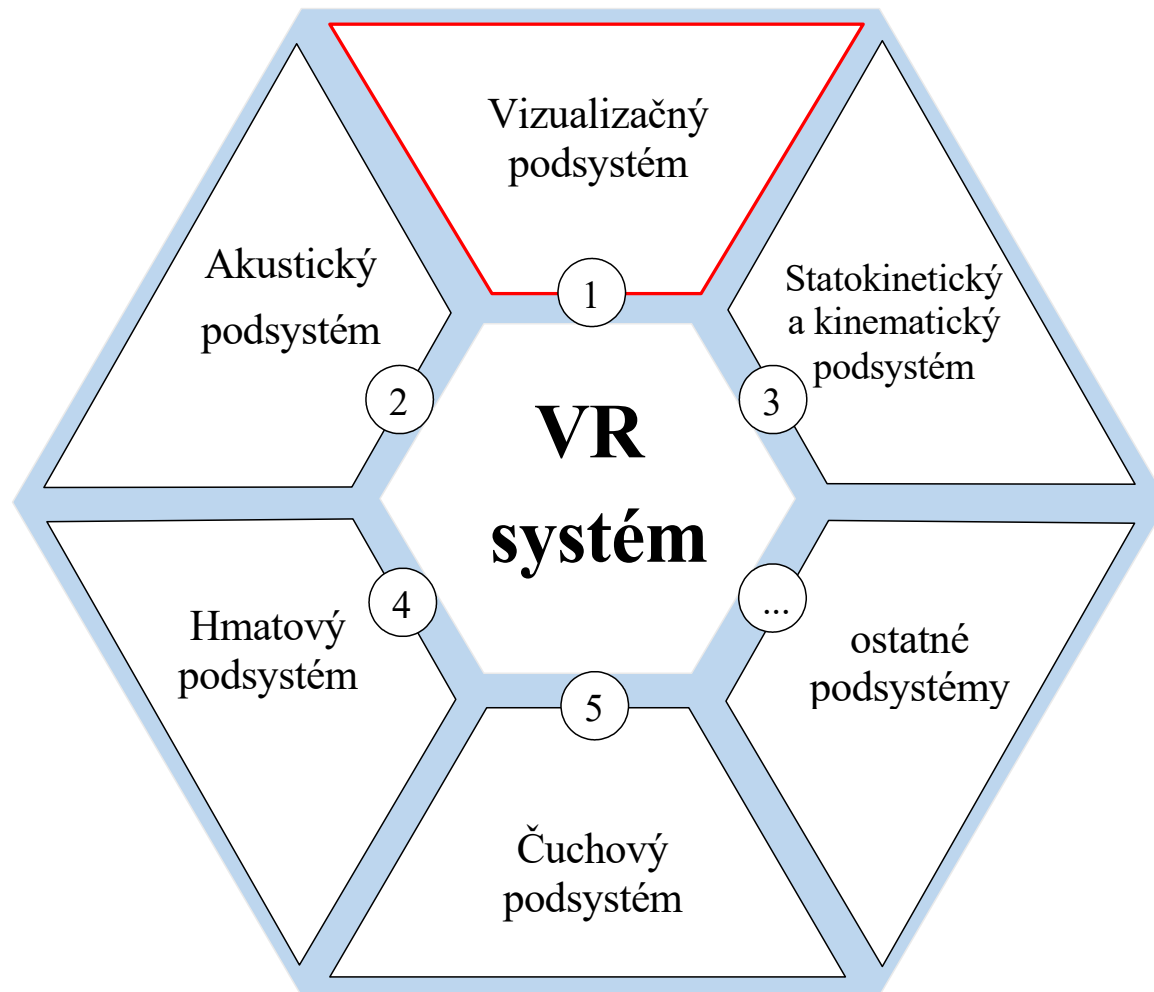
KATEGORIZÁCIA VR SYSTÉMOV

- podľa úrovne V/V prvkov
 - Entry VR
 - Basic VR
 - Medium VR
 - Immersive VR
- podľa dynamiky pozorovateľa a prostredia
 - SESO – Static Environment Static Observer
 - DESO – Dynamic Environment Static Observer
 - SEDO – Static Environment Dynamic Observer
 - DEDO – Dynamic Environment Dynamic Observer



PODSYSTÉMY VR SYSTÉMU

A VZŤAH K POČÍTAČOVEJ GRAFIKE



ZMIEŠANÁ REALITA (ZR/MR)

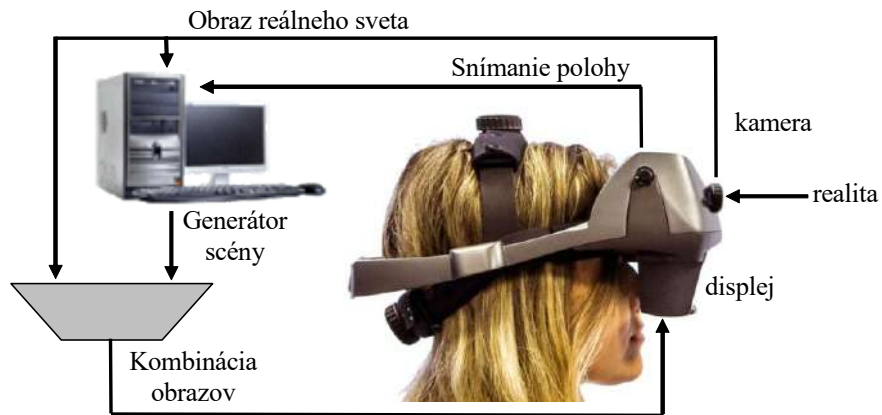
Zmiešaná realita (mixed reality, MR) je oblasť počítačového výskumu zaoberajúca sa kombináciou reálneho sveta a počítačom generovaných dát (virtuálnej reality), kde počítačom generované syntetické objekty sú vmiešavané do reálneho prostredia a naopak, v reálnom čase.

- kombinuje reálne a virtuálne
- je interaktívna
- prebieha v reálnom čase
- je registrovaná v troch dimenziách

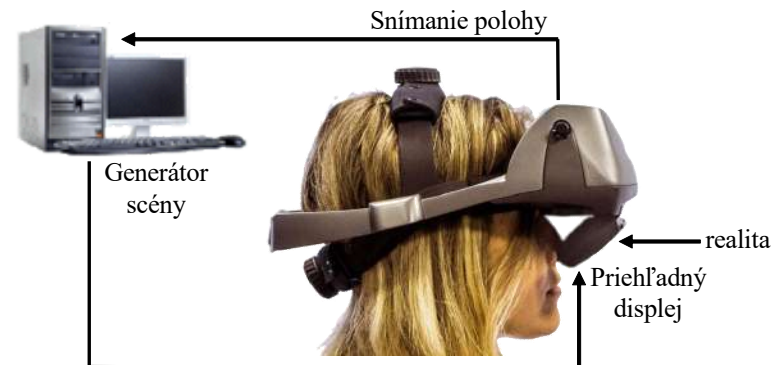


ZMIEŠANÁ REALITA (ZR/MR)

TYPY MR PODĽA TOHO AKO JE VÝSLEDNÝ VNEM KOMPONOVANÝ



*MR s nepriamym (sprostredkovaným)
pohľadom*
(video see-through)
(HMD s bežným/nepriehľadným displejom/i)



MR s priamym pohľadom
(optical see-through)
(HMD s priehľadným displejom/i)



ZMIEŠANÁ REALITA (ZR/MR)

TYPY MR PODĽA SYNCHRONIZÁCIE VIRTUÁLNEJ A REÁLNEJ SCÉNY/OBJEKTOV

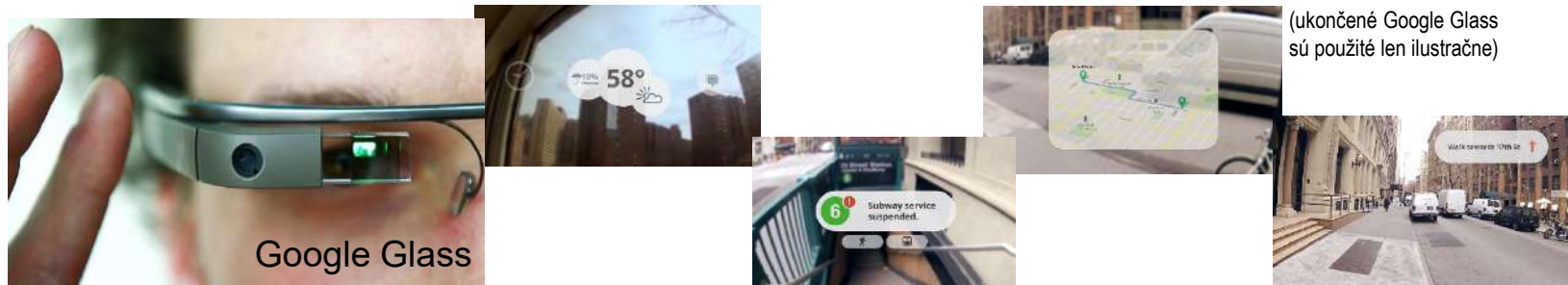
- *systemy s exaktnými značkami* (marked systems) – do reálnej scény sa umiestnia špeciálne značky, ktoré sú počas behu rozpoznané a nahradené virtuálnymi objektmi
- *systemy bez (exaktných) značiek* (semi-markerless, markerless systems) - vyhodnocovanie a vkladanie je bez exaktných značiek, avšak sú potrebné iné doplnkové informácie napr. rozpoznanie obrazu, tváre príp. GPS alebo Wifi signál , 3D sken v reálnom čase a pod.



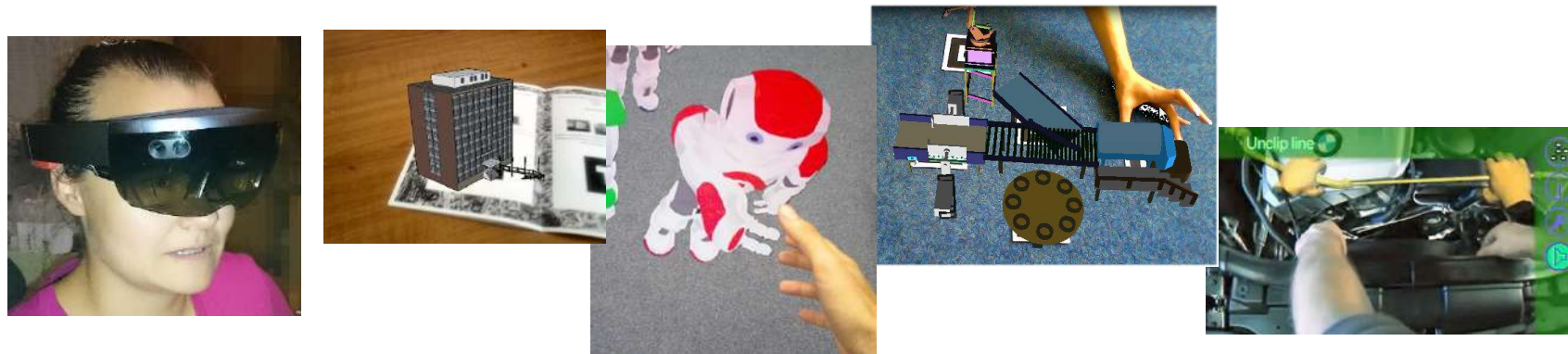
ZMIEŠANÁ REALITA (ZR/MR)

TYPY MR PODĽA TOHO AKO V PRIESTORE DOCHÁDZA K SYNCHRONIZÁCII

- *Obohatená realita* – bez priamej geometrickej väzby s reálnym svetom



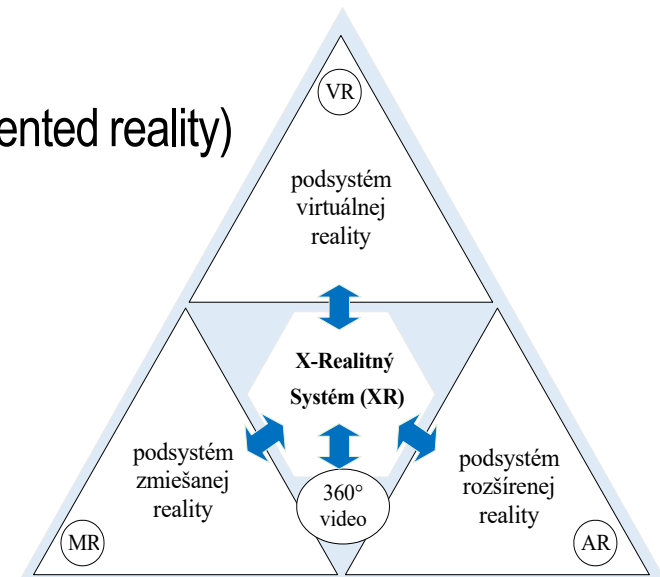
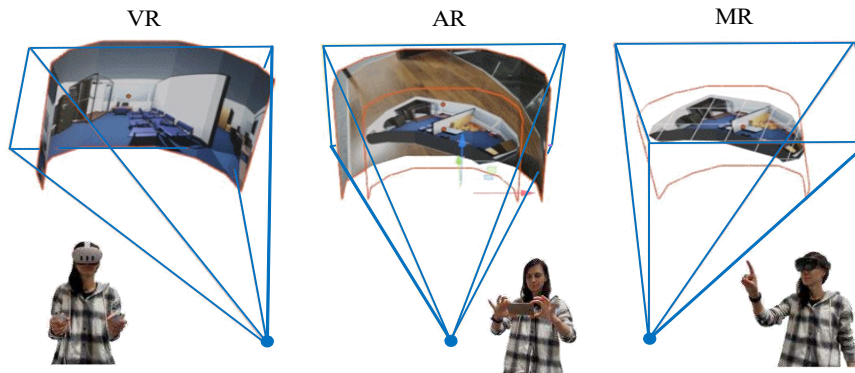
- *Rozšírená realita* – s geometrickou väzbou s reálnym svetom



X-REALITNÝ SYSTÉM (XR)

X-Realita predstavuje inovatívnu oblasť, ktorej účelom je aplikácie zjednotiť technológie :

- *virtuálna realita (VR, virtual reality)*
- *augmentovaná (rozšírená, obohatená) realita (AR, augmented reality)*
- *zmiešaná realita (MR, mixed reality).*



Všetky zložky zdieľajú rovnaký koncept zahrnutia virtuálnych/syntetických objektov/prvkov do reality:

- či už v podobe plne virtuálneho priestoru (napr. virtuálne okuliare (HMD)) – **VR**,
- v podobe premietania virtuálnych prvkov a reality pomocou média (napríklad pomocou kamery) – **AR** a napokon
- v podobe vnímania reality vlastnými očami, kde sa virtuálne prvky projektujú napr. na transparentné displeje alebo inými výstupnými efektmi (Hologramy, okuliare MS Hololens) – **MR**.

vrátane podpory multimodálnosti vstupno/výstupných zariadení.

KOLABORATÍVNA VR A X-REALITA (XR)

System kolaboratívnej virtuálnej reality (CVR) resp. XR (CXR) je založený na báze technológií VR/XR za účelom sprostredkovať spoločne zdieľané virtuálne prostredie medzi viacerými používateľmi v reálnom čase



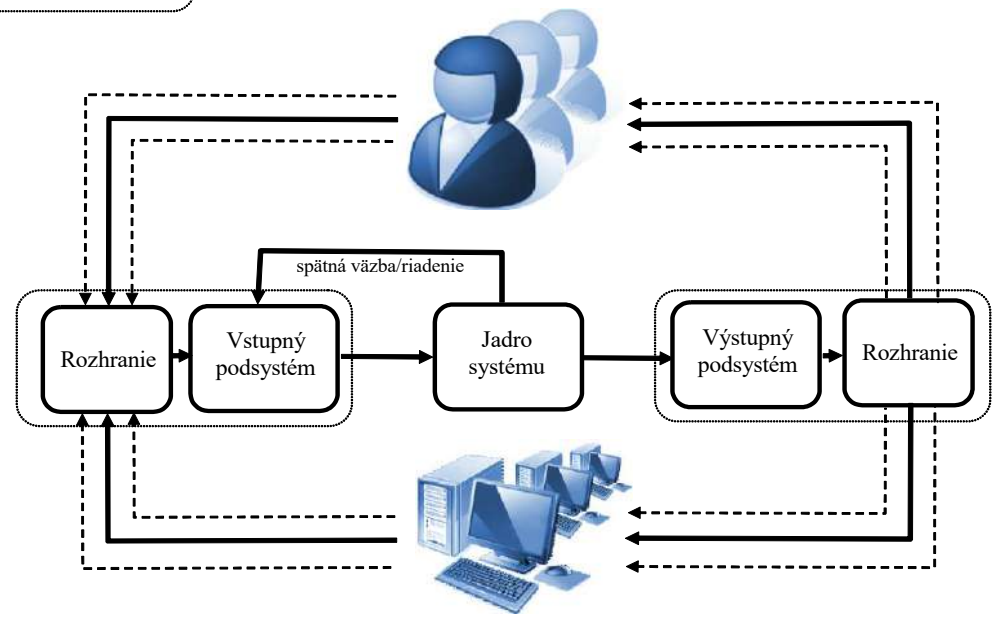
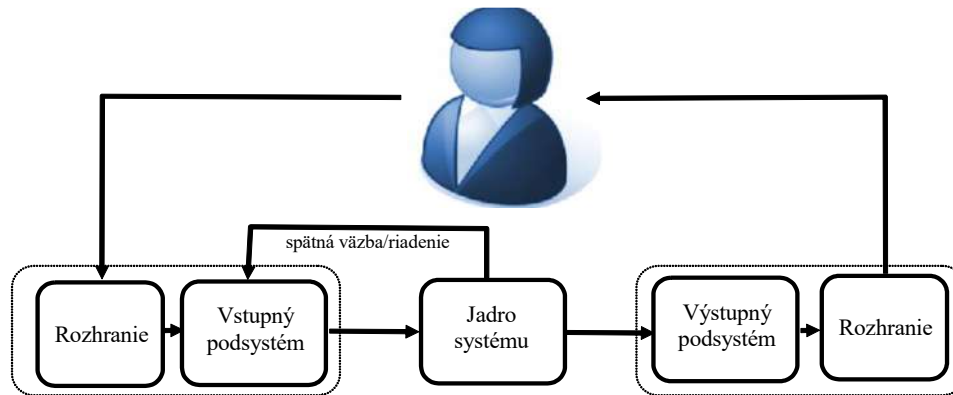
KOLABORATÍVNE VIRTUÁLNE PROSTREDIE

VLASTNOSTI

- Podpora aktivít s integráciou skupiny používateľov.
- Zdieľanie virtuálneho, zmiešaného, alebo zlúčeného kooperatívneho prostredia.
- Zámer - dosahovanie spoločného cieľového riešenia.
- Kombinácia viacerých vstupov v reálnom čase.
- Interaktívna komunikácia a operovanie v zdieľanom kooperatívnom prostredí.



SYSTÉM INTERAKCIE (JEDNO A VIACPOUŽÍVATEĽSKÝ)



POUŽÍVATEĽSKÉ ROZHRANIA A PG (VR)

- Priame, exaktné, mechanické
- Znakové (príkazový riadok (command line), terminál)
- Grafické
- Bio-adaptované, prirodzené a technológie virtuálnej reality



```
Microsoft Windows XP [Verzia 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Bronto>dir
Zoznak v jednotke C nemá žiadnu menovku.
Sériové číslo zväzku je 9853-T716

Úpis adresára C:\Documents and Settings\Bronto
21.08.2008 21:03 <DIR> .
21.08.2008 21:03 <DIR> ..
30.06.2008 23:05 <DIR> Bluetooth Software
25.08.2008 20:59 <DIR> Desktop
03.10.2008 07:27 <DIR> 6 370 Errorlog.txt
15.08.2008 22:24 <DIR> Favorites
16.07.2008 08:24 <DIR> My Documents
20.07.2008 16:05 <DIR> 600 PUTTY.BND
30.06.2008 15:44 <DIR> Start Menu
                2 súborov, 6 370 bajtov
                7 adresárov, 31 000 132 608 voľných bajtov

C:\Documents and Settings\Bronto>
```



LABORATÓRIUM LIRKIS

Unikátne pracovisko na výskum a vývoj nových flexibilných a inteligentných rozhraní systémov na báze počítačovej grafiky a technológií virtuálnej reality



Laboratórium LIRKIS

Laboratórium Inteligentných Rozhraní

Komunikačných a Informačných Systémov

KPI FEI TU Košice

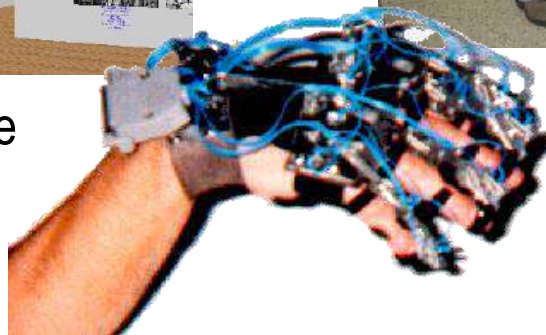
VIRTUÁLNA REALITA A KPI



Pracovisko VR 2. generácie s virtuálnou jaskyňou
(KPI FEI TU Košice)

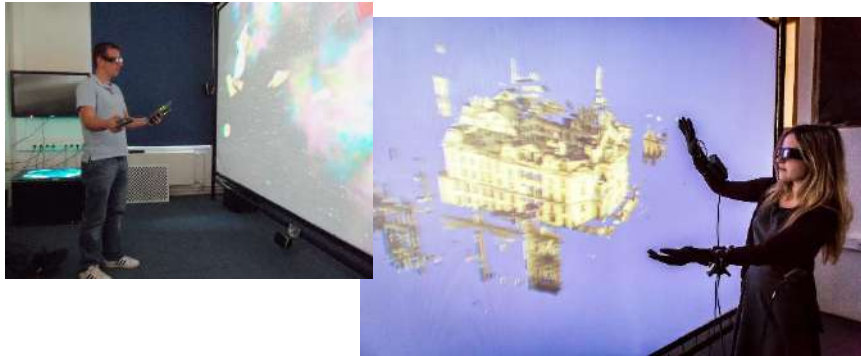


Pracovisko VR 1. generácie
(KPI FEI TU Košice)

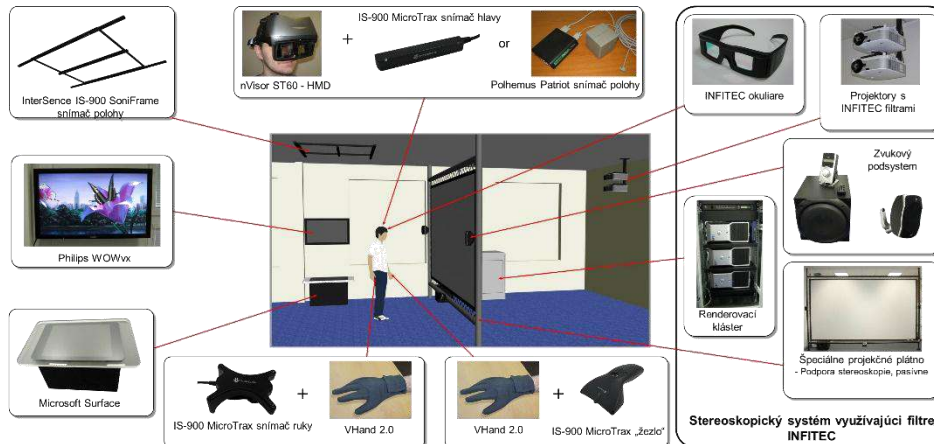


1.-generačná datová rukavica
(KPI FEI TU Košice)

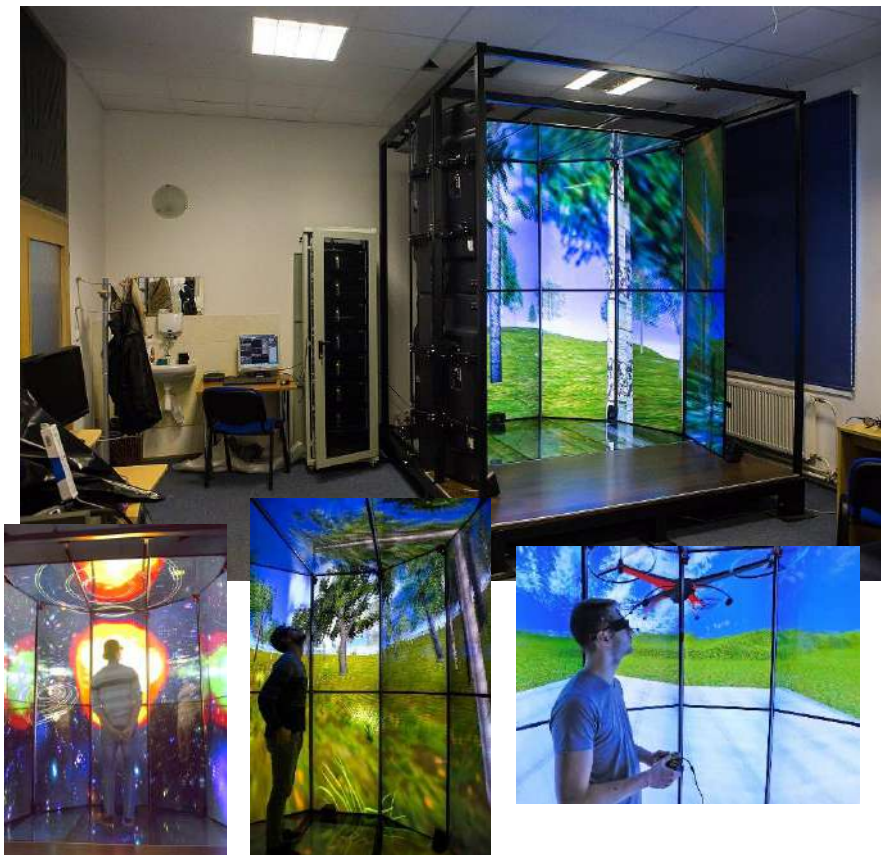
VIRTUÁLNA REALITA A KPI



Pracovisko VR 4. generácie s virtuálnou jaskyňou (LIRKIS KPI FEI TU Košice)



Pracovisko VR 3. generácie (LIRKIS KPI FEI TU Košice)



ZMIEŠANÁ REALITA A KPI

Pracovisko ZR 3. generácie s MS Hololens (LIRKIS KPI FEI TU Košice)



Pracovisko ZR 1. a 2. generácie (LIRKIS KPI FEI TU Košice)

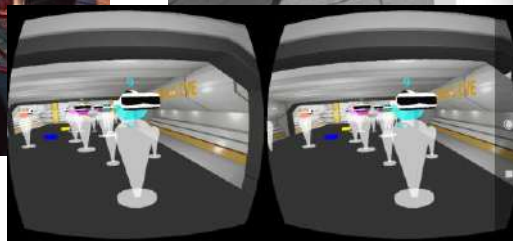


KOLABORATÍVNA VR, X-REALITA A KPI

PRACOVISKO LIRKIS G-CVE 1. GENERÁCIE (KPI FEI TU KOŠICE)



LIRKIS (KPI FEI TU Košice)



Multiplatformové pracovisko na báze LIRKIS G-CVE

Q & A

branislav.sobota@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2025