

Inteligentnejšie riešenie na interpoláciu väčšieho množstva bodov ako polynómy vysokého stupňa?

- Napr. splajny (spliny)
- Čo sú to splajny?
 - Po častiach polynomicke funkcie
 - napr. použijeme polynóm 3. Stupňa a budeme striehnuť na vlastnosti výsledku ...
- Mimochodom uvedený problém s polynómami vyšších rádov sa volá Rungeho fenomén (objavil ho v r. 1901)

Spline funkcie

Spliny sú po častiach polynomicke funkcie daného stupňa s plynulým prechodom medzi jednotlivými časťami. B-Spline $\varphi_{SM}(t)$ stupňa M je tvorený M násobnou konvolúciou „Box“ funkcie:

$$B(t) = \begin{cases} 1 & t \in \langle 0, 1 \rangle \\ 0 & \text{inde} \end{cases}$$

a má kompaktnú podporu na intervale $\langle 0, M+1 \rangle$, $M-1$ spojitých derivácií.

Platí: $\varphi_{S0}(t) = B(t) = \varphi_{Haar}(t)$, takže $\varphi_{S0}(t)$ môžeme generovať pomocou koeficientov $h(n) = (1, 1)$ resp. ich N -násobnými konvolúciami. To odpovedá

Pascalovmu rojuholníku na určenie kombinačných čísiel. Koeficienty $h(n)$ pre B-spline funkcie mierky sú potom:

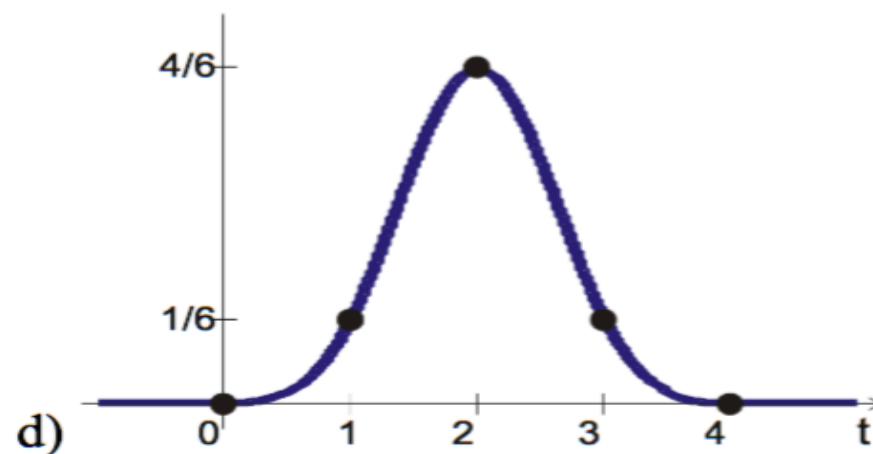
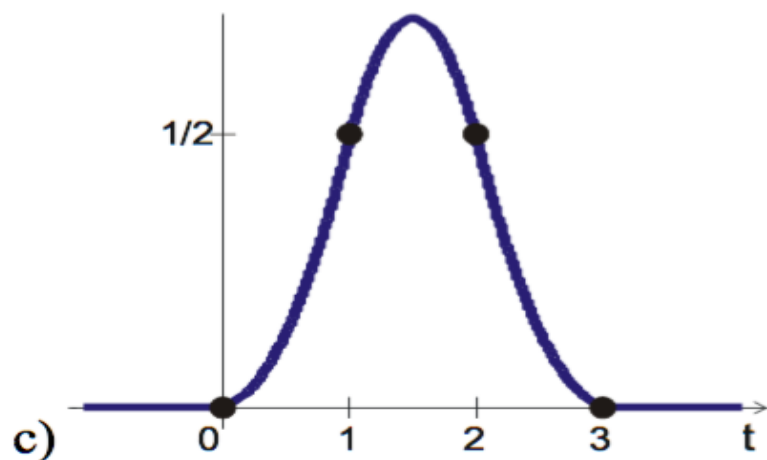
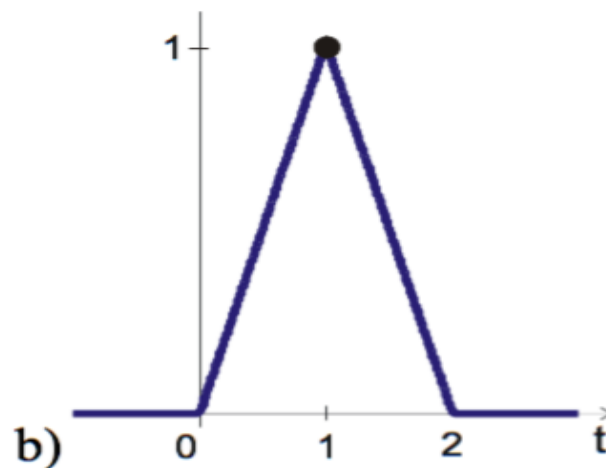
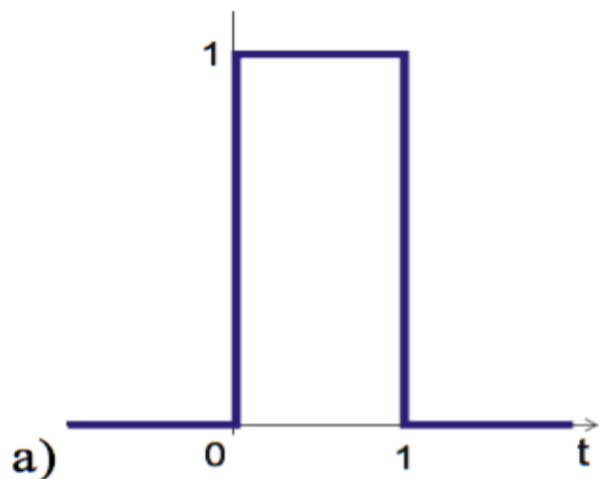
$$h_{S0}(n) = (1, 1) \frac{\sqrt{2}}{2} \quad - \text{konštantný (spline nultého rádu)}$$

$$h_{S1}(n) = (1, 2, 1) \frac{\sqrt{2}}{4} \quad - \text{lineárny (spline prvého rádu)}$$

$$h_{S2}(n) = (1, 3, 3, 1) \frac{\sqrt{2}}{8} \quad - \text{kvadratický(...)}$$

$$h_{S3}(n) = (1, 4, 6, 4, 1) \frac{\sqrt{2}}{16} \quad - \text{kubický(...)}$$

Konštantný, lineárny, kvadratický a kubický spline:



Uzavretý analytický tvar pre kubický B-spline:
[BSPLINE_TVAR]

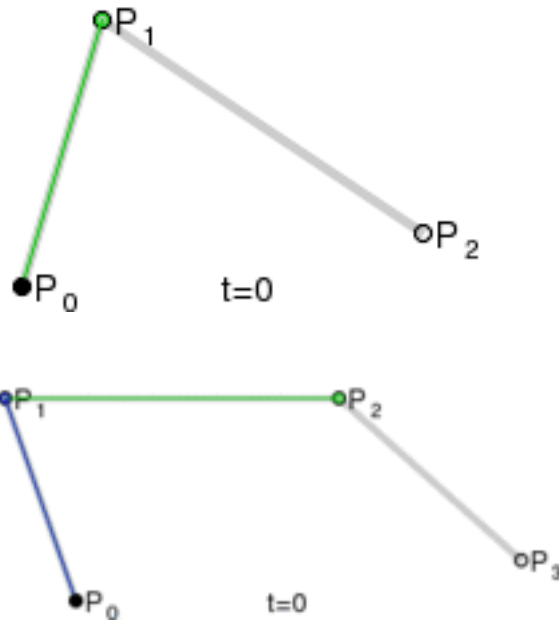
$$h(x) = \frac{1}{6} \begin{cases} 3|x|^3 - 6|x|^2 + 4 & 0 \leq |x| < 1 \\ -|x|^3 + 6|x|^2 - 12|x| + 8 & 1 \leq |x| < 2 \\ 0 & 2 \leq |x| \end{cases}$$

Interpolácia kriviek? Pomocou kubických spline funkcií

- zabezpečia, že
 - výsledný signál je spojitý
 - jeho prvá derivácia je spojitá
 - jeho druhá derivácia $s''(x)$ je spojitá
 - To zabezpečuje že $\int_{x_1}^{x_n} |s''(x)|^2 dx$ je minimálne, t.j. má maximálnu hladkosť, rýchlosť zmeny zmeny fluktuuje minimálne
- parametre kubického spline sa počítajú pre jednotlivé intervaly (susedné dvojice bodov signálu) – iný prístup k návrhu ako pri návrhu interpolačného konvolučného jadra
 - existuje práve jedna priamka ktorá pretína ľubovoľnú dvojicu bodov
 - existuje nekonečne veľa parabol, ostáva 1 stupeň voľnosti
 - existuje nekonečne veľa kubických funkcií ostávajú 2 stupne voľnosti
 - existujúce stupne voľnosti sa využívajú aby bola rovnaká prvá aj druhá derivácia pri nadväzovaní kriviek pre susedné intervaly
 - pri n interpolovaných bodoch je potrebné vyriešiť sústavu n lineárnych rovníc o n neznámych
- Prirodzený (natutal) kubický spline – má pri krajných bodoch zvolenú druhú deriváciu rovnú 0 (t.j. je to rovná čiara pod nejakým uhlom)

Aproximácia bodov

- Viete rozdiel medzi interpoláciou a aproximáciou?
- Beziérove krivky (vymyslel ich Pierre Bezier pre automobilku Renault)
 - lineárne
 - kvadratické [_BEZIER]

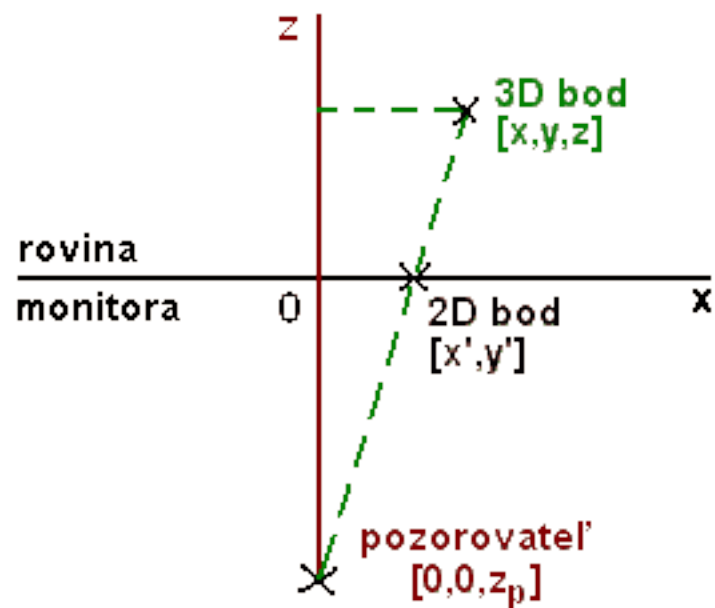


- kubické [_BEZIER]

- B-Spline – používajú sa aj na aproximáciu
- NURBS (nonuniform rational basis spline) zovšeobecnenie Beziérových a B-Spline kriviek
- **A ideme do 3D grafiky ...**

Projekcia z 3D do 2D

- Ak chceme robiť 3D grafiku, ide o základný problém, zobrazovacie zariadenia sú spravidla 2D
- Základné vzorce vyplývajú z podobnosti trojuholníkov



$$\frac{x'}{z_p} = \frac{x}{z_p - z}$$

$$\frac{y'}{z_p} = \frac{y}{z_p - z}$$

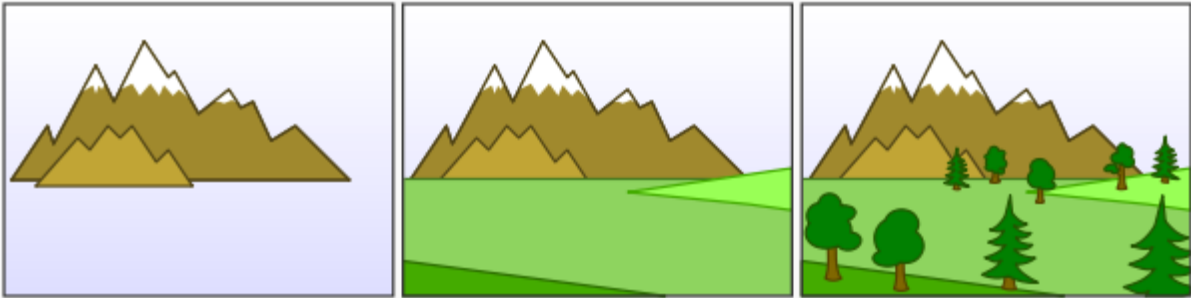
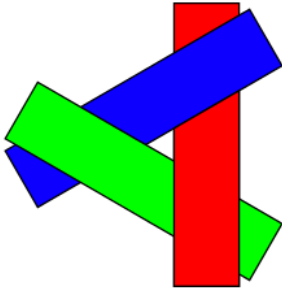
Príklad: Majme bod $B=(3,4,5)$. Aké budú súradnice bodu B' v rovine monitora ak pozorovateľ je vo vzdialenosti $z_p = -1$?

- $x' = z_p x (z_p - z) = -1 * \frac{3}{-1-5} = 0.5$
- $y' = z_p y (z_p - z) = -1 * \frac{4}{-1-5} = \frac{2}{3}$

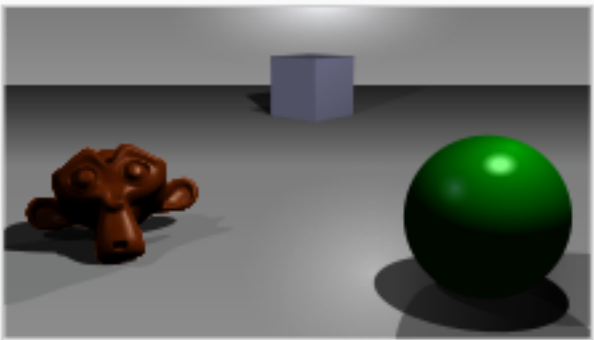
T.j. $B' = (0.5, \frac{2}{3})$

Odstraňovanie neviditeľných hrán/stien

- plochy ktorých kolmice smerujú do opačnej roviny ako je pozorovateľ, tie môžeme vynechať
- **algoritmus maliara:** zotriedime si plôšky podľa vzdialenosti od pozorovateľa, najskôr vykresľujeme tie vzdialenejšie, potom bližšie, ktoré tie vzdialenejšie prekreslia

Tu algoritmus zafunguje			Tu zlyhá ...
			

- **z-buffer:** alokujeme si 2D pole, reprezentujúce rovinu monitoru
 - zaznačujeme si doň vzdialenosť bodu, na ktorý sa v danom smere pozeráme. Ak v danom smere narazíme na bližší bod, tento prekreslí aktuálny a v z. bufferi budeme evidovať jeho vzdialenosť

3D scéna	Obsah z-buffera
	

Mnohosteny

Mnohosten je 3D teleso, ktorého steny sú rovinné mnohouholníky.

Platí eulerova veta pre počty vrcholov (v) hrán (h) a stien (s) konvexného mnohouholníka:

$$v-h+s=2$$

Pravidelné konvexné mnohosteny (existuje ich presne 5) s počtom strán: 4, 6, 8, 12, 20:



Poznámka: útvar je konvexný ak mu patria všetky body na spojnici jeho ľubovoľných 2 bodov.

3D Grafika

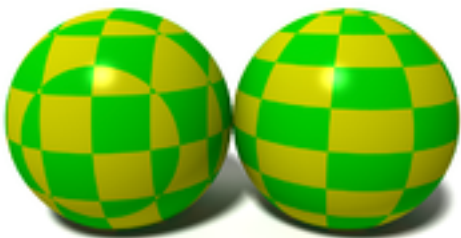
- 4 fázy tvorby obrazu:
 - model
 - osvetlenie
 - materiály
 - rendering

3D Grafika

- Model scény - Objekty
 - Geometria: každé teleso je popísané ako mnohosten, ktorý je úplne určený svojimi hranicami (hrany, strany, stený, vrcholy)
 - Vertex=bod, Edge=hrana, face=plocha
 - Údajová štruktúra, ktorá reprezentuje objekt:
 - CSG konštruktívna geometria pevných telies (modelovanie z primitívnych telies – guľa kváder, valec, kužel, ...) + operácie zjednotenie, prienik, rozdiel.
 - Hraničná reprezentácia –popisuje geometriu hranice na základe vzťahov medzi vrcholmi, hranami a stenami na povrchu
 - Dá sa získať tomografom, 3D skenerom
- Osvetlenie – bodové, plošné, viac zdrojov ...
 - Veľmi dôležité pri vyvolaní dojmu reálnosti
 - všetky algoritmy sú zjednodušením fyzikálnych modelov
 - Phongov model (lokálny model)
 - Radiozia (globálny model)

3D Grafika

- **Materiály** – nastavujú sa vlastnosti materiálov, ktoré ovplyvňujú odraz svetla a ich farebnosť (napr. priehľadnosť, Lesklosť, ...)
 - textúry – obrázok, ktorým je teleso obalené
 - Napr. jednoduchá farba
 - UV mapovanie: textúra v 2D $t(u,v)$ je umiestnená na objekt so súradnicami $p(x,y,z)$, pri mapovaní sa volí projekcia, napr. equirectangular ...



Naľavo bez UV mapovania, vpravo s UV mapovaním

- **Rendering** – je proces, pri ktorom sa z modelu stáva 2D obraz. Je to komplexný proces, pri ktorom vystupuje veľa algoritmov, pri ktorých vystupujú poloha pozorovateľa a osvetlenie a zohľadňujú: usporiadanie objektov v scéne, ich viditeľnosť, farbu, textúru a kvalitu povrchu, optické vlastnosti materiálov, tieňovanie, šírenie svetla, nepriame osvetlenie, tiene, projekciu, potláčanie aliasingu textúr a hrán objektov a mnohé iné.

Rendering

○ príklady

- Tieňovanie, tiene, mäkké tiene – kolísanie farby a jas v závislosti na osvetlení a (čiastočné) prekrytie zdroja svetla iným objektom
- bump mapping – simulácia drobných nerovností na povrchu
- Hmla – simulácia svetla pri prechode atmosférou s hmlou
- odraz svetla (reflexia), lom svetla (refrakcia)
- priehľadnosť /priesvitnosť
- nepriame osvetlenie – osvetlenie odrazeným, resp. lomeným svetlom
- hĺbka ostrosti
- pohybové rozostrenie
- antialiasing

○ algoritmy

- lokálne
- globálne
 - ray tracing – algoritmus spätného sledovania lúčov
 - radiozita – metóda globálneho osvetľovania – má lepšie výsledky ako lokálne osvetľovacia metóda. Využíva fyzikálne zákony o šírení energie v prostredí. Vhodná k simulácii nepriameho (odrazeného) svetla v scéne s matnými povrchmi. Nedokáže pracovať s priehľadnými objektami a zrkadlami.

Anti aliasing

- zamedzuje moire - je ho treba aplikovať pri použití textúr
- základný (a pomalý) algoritmus je FSAA (full scene anti aliasing)
- Používa sa 2xFSAA až 16xFSAA. Aký je postup:
 - o scéna sa najprv renderuje pri $M \times$ väčšom rozlíšení
 - o následne sa výsledný 1 pixel pri $N \times$ FSAA vypočíta ako priemer z N okolitých pixelov
- Príklad pre 8xFSAA:



V súčasnosti existuje viacero druhov AA, aj rýchlejších: FXAA, SMAA, MSAA, SSAA, TXAA, ... (vid'. napr. <https://vr.arvilab.com/blog/anti-aliasing>)

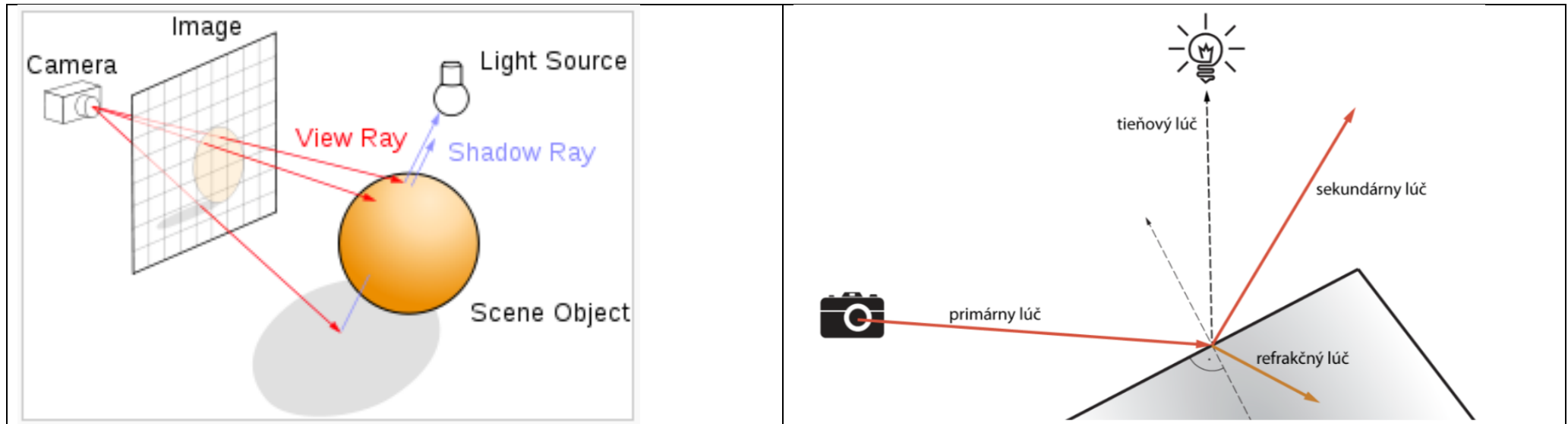
Mapovanie textúry

- Ktorá časť textúry sa použije v akom bode geometrie?
- Pixely textúry=texely
- Mapovanie texelov na pixely scény
 - metóda najbližšieho suseda
 - bilineárna filtrácia
 - trilineárna filtrácia
 - využíva MIP (Multum In Parvo) mapy: spolu s originálnou textúrou sú distribuované predpočítané 2,4,8 x zmenšené verzie (bilineárnou interpoláciou)
 - tretia lin. Interpolácia je so susedných MIP máp nad ktorými boli použité bilineárne interpolácie
 - anizotropné filtrovanie (efektívne aj pri aplikovaní textúr pod veľkým uhlom)
 - trilineárne filtrovanie, ale sú využité t.z.v RIP mapy



Ray tracing

- algoritmus spätného sledovania lúčov
- prečo to počítame naopak? (akoby lúče vychádzali z kamery, nie zo zdrojov svetla) – počítame iba informácie, ktoré vchádzajú do oka



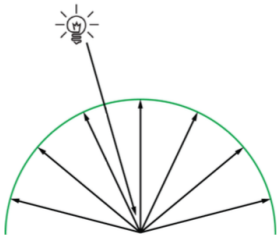
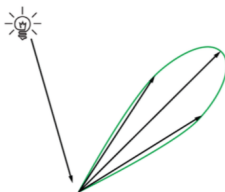
4 typy lúčov: primárne, sekundárne (odrazené), refrakčné (pri transparentných objektoch), tieňové lúče – vedú k svetelným zdrojom, slúžia na výpočet osvetlenia

Algoritmus:

- výpočet sa opakuje pre každý bod priemetne (monitora)
- hľadá sa priesečník polpriamky z oka, cez bod priemetne po prvý priesečník s modelom
- v ňom sa vypočíta lokálne osvetlenie pomocou tieňových lúčov (napr. pomocou Phongovho modelu)
- pokračuje sa rekurzívnym výpočtom intenzity svetla prichádzajúce od refrakčného a sekundárneho lúča

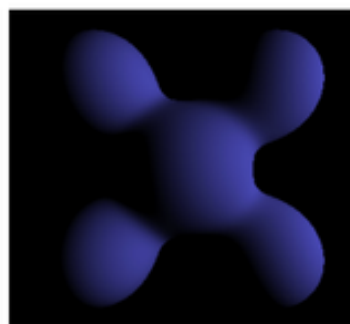
Phongov osvetľovací model

Výsledná intenzita svetla (farba) konkrétneho bodu v scéne je súčtom troch zložiek svetla – ambient (nepriame osvetlenie), diffuse (odrazené a rozptýlené svetlo od matných materiálov), specular (lesklé materiály, ostrosť odleskov od povrchu): $I = I_a + I_d + I_s$

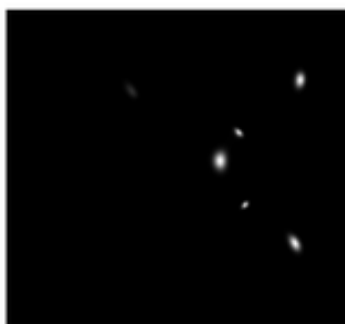
$I_a = k_a L_a$	
$I_d = \frac{k_d}{a - bd - cd^2} (d \cos(\Theta_i)) L_d$ - tlmené konštantným a kvadratickým faktorom	
$I_s = k_s L_s \cos^\alpha(\phi)$	



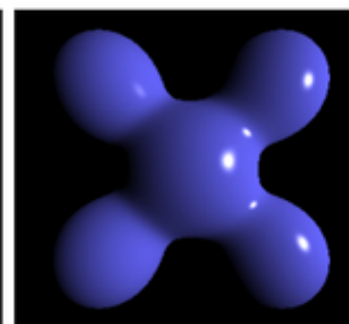
Ambient



Diffuse



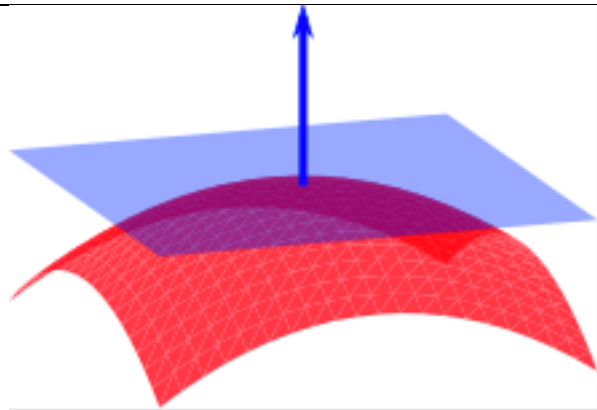
Specular



= Phong Reflection

Tieňovanie

- označuje aplikovanie svetelného modelu na objekt
- veľkú úlohu zohráva smer kolmíc k jednotlivým plôškam
- základné druhy
 - Phongovo tieňovanie – kolmice na jednotlivé plošky sa interpolujú, pomocou osvetľovacieho modelu počíta farbu pre každý bod
 - Gourardove tieňovanie – vypočíta farbu iba vo vrcholoch, medzi tým farby interpoluje
 - ploché tieňovanie vypočíta len jednu farbu a nič neinterpoluje



Hore: kolmica na plochu v bode

Vpravo: smerom doprava - ploché, Gourardove, phongove tieňovanie



Ľudský vizuálny systém a sekvencia obrázkov

- vnímame očami ...
- ako sa správajú tyčinky a čapíky v čase ?
- keď jeden je schopný snímať signál 10x/s sú 2 schopné snímať 20x/s? ... výskum LVS (ľudského vizuálneho systému) prebieha, niektoré otázky sú zodpovedané niektoré nie ...
- Počet snímok za sekundu (fps)
 - koľko fps je potrebných, aká je ale presne otázka? [[100fps](#)]
 - Koľko obrázkov za sekundu vie ľudské oko zbadáť
 - Koľko obrázkov za sekundu potrebujem, aby pohyb bol plynulý?
 - Koľko obrázkov za sekundu treba aby obraz „neblikal“?
 - Aký je najkratší čas, ktorý ešte ľudské oko rozozná?
 - Vyššie uvedené otázky zďaleka neznameniajú to isté!
- Ak je napr. fps=18 a obrázky sú pohybovo správne rozostrené, je pohyb vnímaný plynulejšie, ako keď máme 50fps ale obrázky sú dokonale zaostrené
- Pri bielom obraze stmavnutie dlhé 1/100 s nezbadáme
- Pri čiernom obraze zasvietenie dlhé 1/220s ešte zbadáme aj rýchlejšie ... (dopadajú fotóny)
- Takže super ilúziu spojitosti obrazu asi budeme možno mať až pri 500Hz monitoroch / televízoroch.
- Mimochodom: Sme schopní vidieť 1 fotón? (hoci sietnica je schopná ho zaregistrovať neznamená to, že sa vyšle signál do mozgu. Ten sa vyšle ak takýchto udalostí je 5-9 za 100ms

Video= Video signal, či viac?

- Video = video signál a celá technologická oblasť ktorá s ním súvisí
 - Video zložka môže byť 2D, 3D
 - obsahuje aj audio, zvuk môže byť stereo, 3D,
- Vznik
 - Snímanie: a) reálny svet – kamera b) už s digitálneho zdroja?
 - vytvorenie
 - animácia
 - statické texty, obrázky, ...
 - kompozícia zdrojov audio/video – sériovo/paralelne
- Prenos/šírenie/uloženie
 - Streaming/broadcast - živý/zo záznamu?
 - Realtime obojsmerné?
 - Štúdiová kvalita? Šírenie konečnému spotrebiteľovi?
- Spotreba/konzumácia/prezeranie
 - (3D) Monitor? Videostena? (3D) Projektor/Kino? 3D 360°kino?
 - 3D okuliare?
 - VR= virtuálna realita (OculusRift, ...)

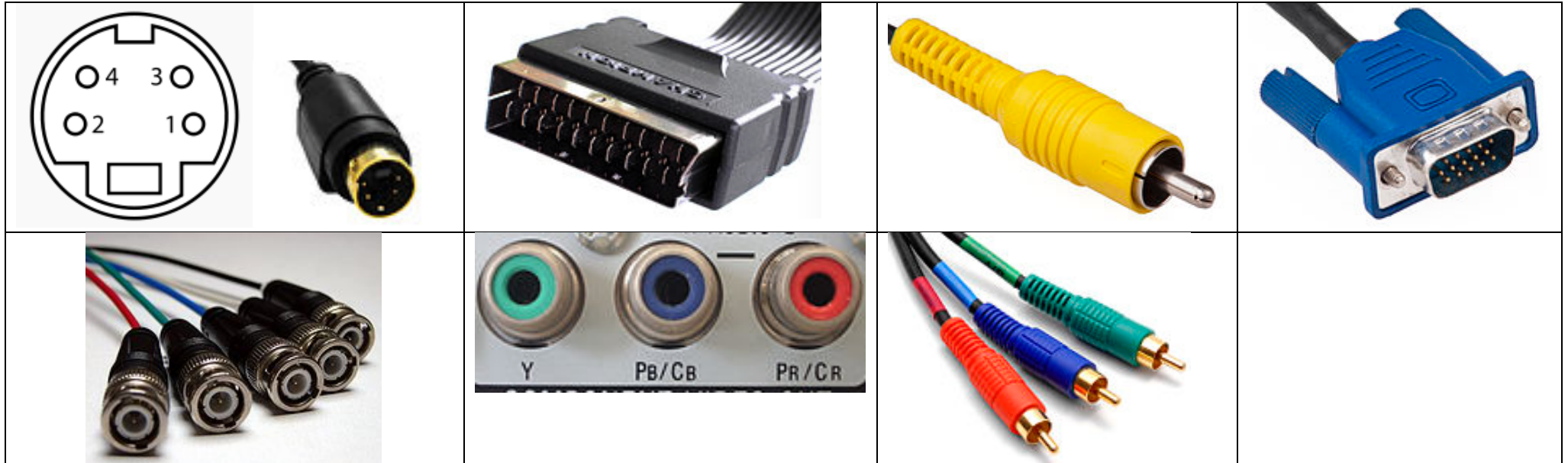
Snímanie videa

- Analógové formáty
 - PAL - Phase Alternating Line, QAM, (25p 50i), 720x576
 - NTSC – National Television System Committee (30p 60i), 720x480
 - SECAM - Séquentiel couleur à mémoire
 - u nás bol kedysi SECAM-B/G
- Koľko fps?
 - Priemyselné vysokorýchlostné kamery (pre rýchle deje) cca 7000 fps
 - „time lapse“ kamery – pre pomalé deje (podporujú aj bežné smartfóny
 - bežné kamery podporujú 35, 30, ...240fps
 - keď sa robí screen recording, stačí 5 fps ... (samozrejme záleží od cieľových požiadaviek)

Video signály

Analógové štandardy (ktoré z nich obsahujú aj audio?)

- kompozitné video (BEZ AUDIA) – priamo obsahuje NTSC/PAL/SECAM signál ...
- S-video (dvojkanálové YC) (BEZ AUDIA)
 - Lepšia kvalita kvoli oddeleniu jasového(Y) od farebného(C) signálu
- Componentové video (trojkanálové: YPbPr ($Y = 0.2126 R + 0.7152 G + 0.0722 B$, $Pb = B - Y$, $Pr = R - Y$), alebo RGB), BEZ AUDIA
- SCART (obsahuje kompozitný, RGB, YpbPr) , S AUDIOM
- VGA (RGBHV) – H=Horizontal sync, V=Vertical Sync), BEZ AUDIA



Digitálne štandardy

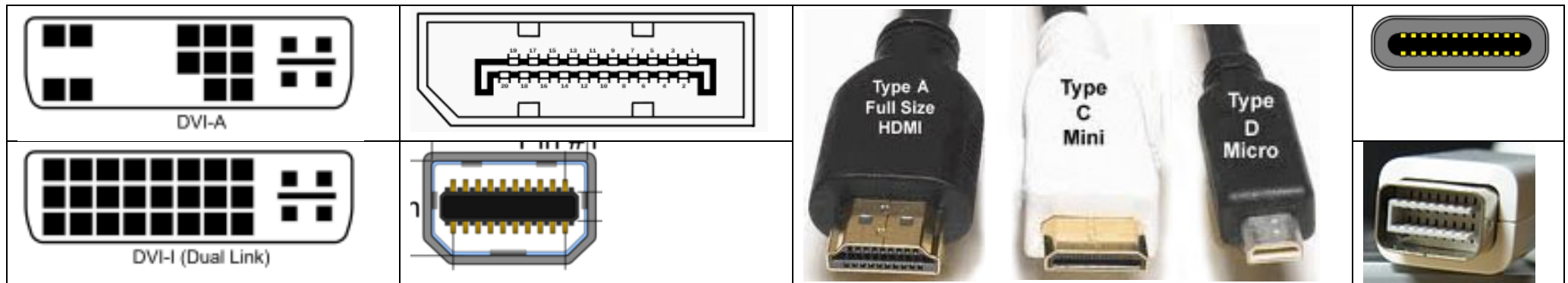
- **DVI** – prenáša aj analógový VGA signál (ako jediný), neprenáša AUDIO
 - DVI-I = analog aj digital
 - single link do 1920x1200 pri 60Hz.
 - dual link do 2560x1600 pri 60Hz.
 - DVI-D = len digital
 - DVI-A = len analog
 - Digitálny signál sa prenáša 4 kanálmi - 8b10b kódovanie (vyrovnanie DC zložky)
 - 3x komponentný RGB=24 bitov per pixel
 - 4. Kanál je synchronizačný
- **HDMI**
 - Neprenáša analógový signál
 - Prenáša AUDIO (v základe nekomprimované PCM stereo - povinné)
 - voliteľne až Dolby TrueHD ...
 - spätne kompatibilné s DVI (nie je potrebná konverzia signálu) – existujú adaptéry obojsmerni
 - HDMI 1.0 (max freq. Pixelov 165MHz) - do 1920x1200 pri 60Hz.
 - HDMI 1.3 (max freq. Pixelov 340MHz) - do 2560x1600 pri 60Hz.
 - HDMI 1.4 obsahuje HEC=HDMI Ethernet Channel – 100Mbit obojsmerné pripojenie
 - HDMI 2.0 4K pri 60Hz, podpora HDR
 - HDMI 2.1 10K pri 120 Hz (nad 8K sa používa Display stream Compression (DSC))

Display port

- prenáša audio (voliteľne) 16/24 PCM, 32KHz-192kHz vzorkovacia frekvencia
- nekompatibilný s HDMI/DVI
- obsahuje napájanie, takže je možné zostrojiť aktívny konvertor na VGA/HDMI/DVI
- verzia 1.2 – 4096x2160 pri 60hz
- verzia 1.3 – 5120x2880 pri 60hz
- verzia 1.4 – 7680x4320 pre 60Hz, podporuje DSC

USB typ C

- Podporuje alternatívne módy – čo majú robiť dedikované piny
 - Konfiguruje sa pomocou VDM (vendor defined messages)
 - Podporuje: DisplayPort, MHL, Thunderbolt, HDMI



DSC (Display Stream Compression)

- VESA štandard (vid'. <https://glenwing.github.io/docs/VESA-DSC-1.2a.pdf>)
- Používa vizuálne bezstratový kompresný algoritmus
- Založené na delta PCM (presnejšie adaptívna predikcia pomocou mediánu) + aritmetické kódovanie (vid'. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=112371>)
- Používa YCoCg (luminancia, chrominancia zelená, chrominancia oranžová) priestor
 - lepšie dekorelovanie farebných zložiek
 - možné rýchle prepočítanie z RGB modelu

$$\begin{bmatrix} Y \\ Cg \\ Co \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \\ \frac{1}{2} & 0 & -\frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & -1 \end{bmatrix} \cdot \begin{bmatrix} Y \\ Cg \\ Co \end{bmatrix}$$

- YCoCg-R (reversible) – bezstratové prepočítanie - používa **lifting schému**:

```
Co  = R - B;      tmp = Y - Cg/2;
tmp = B + Co/2;    G   = Cg + tmp;
Cg  = G - tmp;     B   = tmp - Co/2;
Y   = tmp + Cg/2;  R   = B + Co;
```

HDCP protocol

- Dá sa nelegálne kopírovať digitálny obsah?
- High-bandwidth Digital Content Protection
- Podporované v prípade DVI, HDMI, DP
- Vysielač si overí, či prijímač podporuje HDCP a vyšle ich, iba keď áno a vyšle ich zašifrované aby ich niekto neodchytil.
 - Každé zariadenie má 40 56bitových kľúčov, z nich je vytvorený verejný 40 bitový kľúč KSV, ktorý má 20 bitov jednotkových a 20 nulových
 - počas fázy autentifikácie si vymenia KSV kľúče pomocou procedúry Blomova schéma
 - výsledkom je, že obaja majú na konci to isté 56 bitové číslo, ktoré je použité na šifrovanie
- následne sa použije streamový šifrátor (stream cipher)
 - každý pixel je XORovaný vygenerovaných 24 bitovým číslom
- v súčasnosti verzia HDCP 2.2 – aj tú sa podarilo hacknut (HDFury) – zariadenie vie odstrániť HDCP a poslať signál ďalej

Snímanie

- Kamery
 - analógové
 - digitálne – už sa stretávame iba s takýmito ...
- v čom sa kamery spravidla odlišujú od fotoaparátov?
 - optická stabilizácia
 - rýchlosť zaostrovaní – kontinuálne
 - prispôsobenie expoziície – kontinuálne
 - záznam audia
 - kapacita média

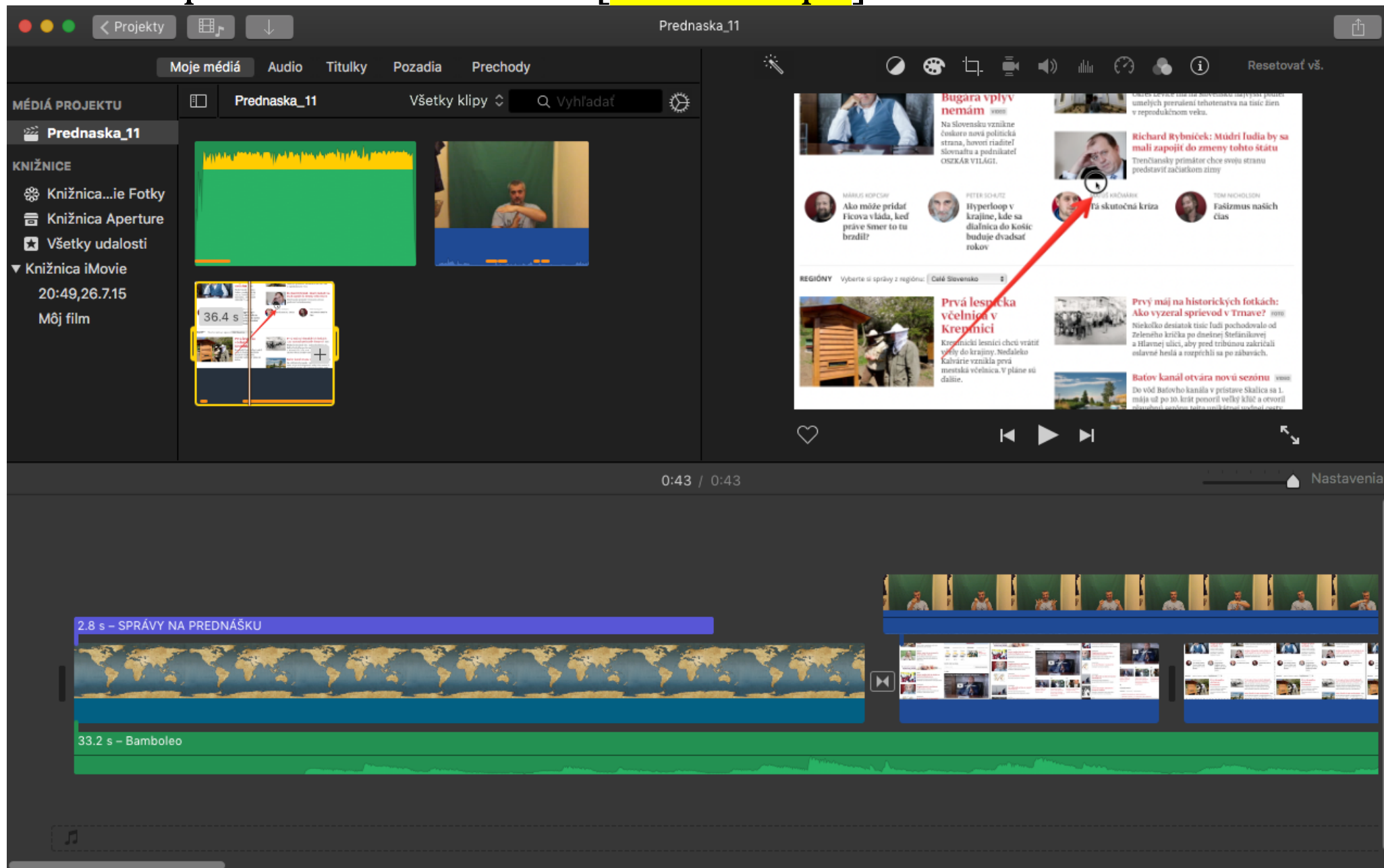
Spracovanie

- video mixpult ...
 - iMovie, FinalCut Pro
 - Windows Movie Maker ...

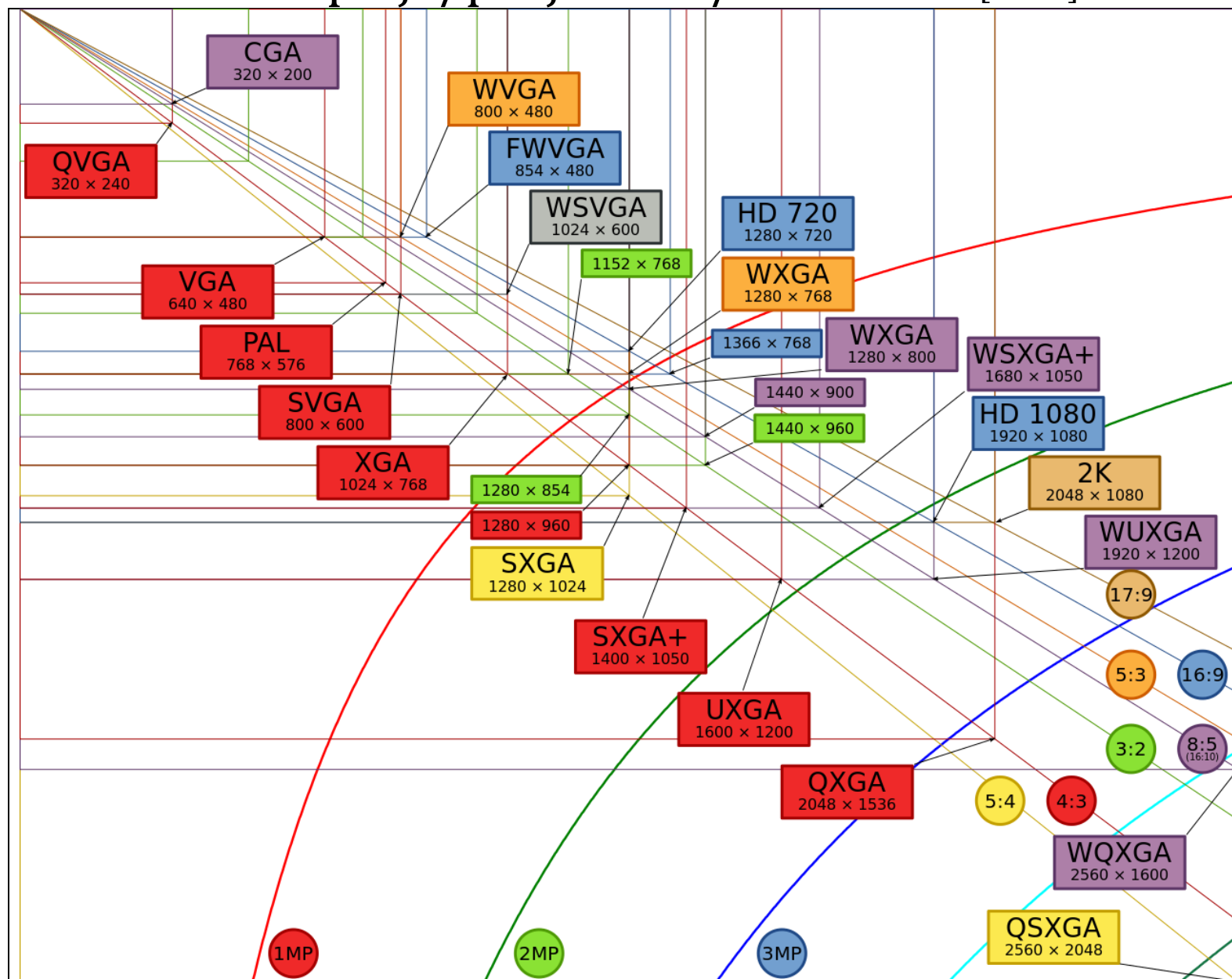
Publikovanie

- Youtube?
- iCloud?
- ...

Príklad spracovanie iMovie + [ukazka.mp4]



Rozlíšenie displeja/projektora/televízora [www]



- Pomery strán od 5:4 po 17:9 (sklon čiar na obrázku)
- 2K, 4K, ... určuje to **koľko majú cca. stĺpcov** (napr. 2K=2048x1080, 4K=3840x2160, 8K=7680x4320)
- Z hľadiska televízie
 - SD=PAL/NTSC
 - HD=HD720, HD1080 (**čísla označujú počet riadkov**)
 - UHD=4K, 8K
- Obrázok by mohol pokračovať až po 8192x8192 (8K Fullframe, 67Mpix)

