

SVR 2025/26 – vypracované okruhy na skúšku

Vypracované okruhy k písomnej skúške zo Systémov virtuálnej reality (SVR). Zamerané na kľúčové informácie z prezentácií, spracované pre lepšie učenie.

1) Virtuálna realita, historické nadväznosti, základné pojmy, atribúty a technológie, ľudské limity, avatar.

1. Virtuálna realita (VR): Definícia a atribúty

Virtuálno-reálny systém je interaktívny počítačový systém, ktorý vytvára ilúziu syntetizovaného priestoru alebo dokonalú simuláciu prostredníctvom tesného spojenia človeka a výpočtového systému.

Medzi **základné atribúty VR** patria:

- **Imerzia:** Pohltenie používateľa do prostredia a jeho emocionálne vnímanie (pozor, imerzia sa nerovná komfortu).
- **Percepcia:** Prítomnosť používateľa v prostredí a jeho dimenzionálne vnímanie.
- **Interakcia:** Možnosť komunikácie používateľa s virtuálnym prostredím.

2. Historické nadväznosti a osobnosti

História VR siaha hlbšie, než by sa mohlo zdať:

- **1940:** Trenažér torpédového bombardéra (Crail, Škótsko) ako príklad totálnej simulácie s kruhovou projekciou.
- **1960:** **Morton L. Heilig** navrhol *Telespheremask* a neskôr zariadenie *Sensorama* (simulácia jazdy motocyklom cez Brooklyn).
- **1963:** **Ivan Sutherland** vytvoril prvý statický HMD (Head-Mounted Display).
- **Kľúčové osobnosti:** Okrem Heiliga a Sutherlanda sú to Ronald T. Azuma (definícia MR), Jaron Lanier (popularizácia pojmu VR) či Nadja M. Thalmann.
- **Lokálna história:** Na FEI TU v Košiciach pôsobí pracovisko LIRKIS, ktoré prešlo vývojom od 1. generácie (dátové rukavice) až po súčasnú 4. generáciu s virtuálnou jaskyňou (CAVE).

3. Základné pojmy a technológie

Súčasná terminológia rozlišuje rôzne formy reality pod spoločnou skratkou **XR (eXtended Reality)**:

- **Virtuálna realita (VR):** Plne syntetický priestor, spravidla v HMD okuliaroch.
- **Rozšírená realita (AR):** Vkladanie virtuálnych prvkov do reálneho sveta cez médium (kamera mobilu).
- **Zmiešaná realita (MR):** Prepojenie reality a virtuality v reálnom čase, kde virtuálne objekty presne reagujú na fyzické prostredie (napr. HoloLens).

- **Augmentovaná virtualita (AV):** Vkladanie reálnych objektov do virtuálneho sveta (opačný princíp k AR).
- **Metaverzum:** Prepojená sieť 3D virtuálnych svetov pre prácu, hru a sociálny kontakt, postavená na Web 3.0 a blockchaine.
- **HCI (Human-Computer Interaction):** Vedná disciplína zaoberajúca sa návrhom a realizáciou interaktívnych systémov pre človeka.

4. Ľudské limity vo VR

Používanie VR systémov naráža na biologické obmedzenia človeka:

- **Kybernetická nevoľnosť (cybersickness):** Vzniká pri nesúlade medzi zrakovým vnímaním pohybu a vestibulárnym aparátom (rovnováhou), čo spôsobuje bolesť hlavy a únavu.
- **VAC konflikt (Vergence-Accommodation Conflict):** Oči v HMD menia konvergenciu (uhol pohľadu na blízke objekty), ale akomodácia (zaostrenie šošovky) zostáva fixná na rovinu displeja, čo vedie k únave očí.
- **Fyzické a kognitívne limity:** Únava pohybového aparátu, obmedzená ergonómia zariadení a senzorické preťaženie.

5. Avatar a Digitálne dvojča

- **Avatar:** Psycho-morfologická štruktúra, ktorá reprezentuje používateľa vo virtuálnom svete. Má definované vizuálne vlastnosti, schopnosti a implementačnú štruktúru (napr. hlava s kamerou, telo so systémom kolízií).
- **Digitálne dvojča (Digital Twin):** Sofistikovaný "avatar" reálneho výrobku alebo procesu. Slúži na zber dát, simuláciu a optimalizáciu výroby (napr. v Industry 4.0), čím prepája realitu s virtualitou prostredníctvom teleriadenia a virtualizácie.

2) Kategorizácia VR systémov podľa úrovne technických prostriedkov a na základe dynamiky pozorovateľa a prostredia.

1. Kategorizácia podľa úrovne technických (V/V) prostriedkov

Systémy sa delia podľa kvality a komplexnosti vstupno-výstupných (V/V) zariadení, ktoré určujú mieru interakcie a vnemov používateľa. Rozlišujeme štyri základné úrovne:

- **Entry XR (VR):** Vstupné systémy s najnižšou úrovňou technických prostriedkov.
- **Basic XR (VR):** Základné systémy poskytujúce štandardné funkcie.
- **Medium XR (VR):** Stredná trieda systémov s lepšou kvalitou zobrazenia a sledovania.
- **Immersive XR (VR):** High-end systémy navrhnuté pre hlboké pohltenie používateľa (imerziu).

Je dôležité poznamenať, že **vyššia technologická úroveň systému automaticky neznamená vyšší komfort** používateľa.

2. Kategorizácia podľa dynamiky prostredia a pozorovateľa

Táto klasifikácia rozlišuje systémy podľa toho, či sa objekty v prostredí hýbu (dynamika) a či sa môže v priestore voľne pohybovať aj samotný používateľ (pozorovateľ).

Používajú sa štyri základné akronymy:

- **SESO (Static Environment Static Observer):** Statické prostredie aj statický pozorovateľ.
- **DESO (Dynamic Environment Static Observer):** Dynamické prostredie (objekty sa hýbu), ale pozorovateľ zostáva na mieste.
- **SEDO (Static Environment Dynamic Observer):** Statické prostredie, v ktorom sa pozorovateľ môže voľne pohybovať.
- **DEDO (Dynamic Environment Dynamic Observer):** Najkomplexnejší systém, kde je dynamické prostredie aj pohyb pozorovateľa.

3. Detailnejšie delenie komplexných (DEDO) systémov

Pri systémoch typu **DEDO** sa sledujú ďalšie dôležité parametre, ktoré ovplyvňujú výsledný zážitok:

- **Typ prostredia:** Rozlišujeme **Living** (živé, s procedurálnymi animáciami) alebo **Non-living** (neživé) prostredie.
- **Počet avatarov:** Systém môže byť určený pre jedného používateľa (**Single**) alebo viacerých (**Multi**).
- **Riadenie avatarov:** Avatary môžu byť riadené človekom (**Human**) alebo systémom (**Non-human**), pričom ich správanie môže byť prirodzené, umelé alebo kombinované.

3) VR systém a jeho podsystémy, interakcia vo VR systémoch.

1. VR systém a jeho podsystémy

Virtuálno-reálny systém nie je monolitický, ale skladá sa z viacerých špecializovaných podsystémov, ktoré spolupracujú na vytvorení imerzívneho zážitku.

- **Vizualizačný podsystém:** Je kľúčový, pretože sprostredkúva až **80 % informácií**. Zodpovedá za renderovanie grafiky, riešenie viditeľnosti, osvetlenie a kompozíciu obrazu pre výstupné zariadenie (napr. HMD alebo CAVE).
- **Akustický podsystém:** Zabezpečuje priestorový zvuk, simuluje odrazy zvuku a zvuky pri kolíziách objektov.
- **Hmatový (haptický) podsystém:** Sprostredkúva hmatové vnemy, silu spätnej väzby a vibrácie prostredníctvom zariadení, ako sú dátové rukavice.
- **Statokinetický a kinematický podsystém:** Sleduje polohu používateľa a simuluje fyzikálne sily (rovnováha, zrýchlenie), ktoré pôsobia na "cybernauta".
- **Ostatné podsystémy:** Čuchový (vône objektov) alebo bio-senzorické podsystémy (snímanie EEG, tepu).

2. Štruktúra virtuálneho objektu

Každý objekt vo VR systéme (nazývaný aj **asset**) má presne definovanú štruktúru:

- **Globálne parametre:** Spoločné pre celý objekt (poloha v priestore, materiál, rozmery, tvar).
- **Lokálne rámce (Reprezentácia):** Špecifické vlastnosti pre jednotlivé zmysly (farba pre vizualizáciu, zvuk pre akustiku, deformácia pre kinematiku, sila vibrácií pre hmat).
- **Logika a správanie:** Definuje, ako objekt reaguje na podnety a aké má schopnosti.

3. Interakcia vo VR systémoch

Interakcia prebieha v rámci tzv. **slučky interakcie** (interakčnej rúry), kde dáta prúdia v reálnom čase:

1. **Vstupný podsystem:** Zachytáva aktivity používateľa cez rozhrania (ovládače, gestá, sledovanie očí).
2. **Jadro systému:** Vyhodnocuje logiku, rieši kolízie a aktualizuje stav virtuálneho sveta.
3. **Výstupný podsystem:** Generuje spätnú väzbu (obraz, zvuk, haptiku), ktorú používateľ vníma.

4. Základné interakčné techniky

Používateľ vo virtuálnom prostredí vykonáva tri základné druhy aktivít:

- **Selekcia:** Statická (jednorazová) aktivita. Slúži na označenie objektu, zmenu jeho vlastností alebo teleportáciu.
- **Manipulácia:** Dynamická aktivita vykonávaná v čase. Zahŕňa presúvanie, otáčanie, škálovanie alebo deformovanie objektov.
- **Lokomócia:** Dynamická aktivita umožňujúca prirodzený pohyb používateľa (kamery) v rozsiahlom virtuálnom prostredí.

5. Typy rozhraní

Systémy sa snažia o dosiahnutie **naturálnych rozhraní (NUI)**, ktoré sú čo najbližšie k reálnym ľudským vnemom.

- **Naturálne:** Priame využitie tela (gestá rúk, sledovanie pohľadu – eyetracking).
- **Bio-adaptované:** Využívajú biologické signály (EEG dáta, svalové napätie) na riadenie prostredia.
- **Mechanizmy v kolaborácii:** Ak v systéme interaguje viac používateľov naraz, interakcia sa riadi buď **blokováním** (objekt uzamkne prvý používateľ) alebo **voľnou interakciou** (používatelia si môžu konkurovať).

4) X-realita (XR), Hyper-realita, adaptabilita rozhraní XR systémov.

1. X-Realita (XR – eXtended Reality)

X-Realita predstavuje inovatívnu oblasť a **zjednotenú aplikačnú platformu**, ktorej účelom je prepojiť technológie virtuálnej, rozšírenej a zmiešanej reality na úrovni aplikácií.

- **Zložky XR:** Pozostáva z troch základných pilierov: **VR** (plne virtuálny priestor), **AR** (premietanie cez médium/kameru) a **MR** (priame vnímanie reality s projekciou na transparentné displeje). Voliteľne zahŕňa aj prvky 360° videa.
- **Štandardizácia:** Dnes je XR technologicky zastrešená pod priemyselným štandardom **OpenXR** (od Khronos Group), ktorý rieši nejednotnosť API rôznych výrobcov a oddeľuje aplikáciu od konkrétneho hardvéru.

2. Hyper-realita (HR)

Hyper-realita je pokročilý stupeň reality, ktorý vzniká technologickou kombináciou VR, MR a AR, pričom zahŕňa aj **fyzickú kombináciu** prostredí.

- **Príklad z praxe:** Používateľ s dátovou prilbou s transparentnými displejmi (MR) sa nachádza vo vnútri virtuálnej jaskyne (VR).
- **Vzťah k Metaverzumu:** HR sa považuje za technologický základ pre Metaverzum – prepojenú sieť 3D svetov na báze Web 3.0 a blockchainu.

3. Adaptabilita rozhraní XR systémov

Adaptivita je schopnosť rozhrania prispôbiť svoje ovládanie a štruktúru konkrétnemu typu používateľského zariadenia v reálnom čase.

- **Účel:** Zabezpečiť vhodné interakčné techniky podľa toho, či používateľ pristupuje k systému cez mobil, headset alebo desktop.
- **Typy podľa účelu nasadenia:**
 1. **Štandardné adaptívne rozhrania:** Riadia interakciu jednotlivých podsystémov (simulačný, fyzikálny, senzorický).
 2. **Viacpoužívateľské adaptívne rozhrania:** Integrujú multiplatformový prístup, kde v jednom kolaboratívnom prostredí môžu naraz interagovať používatelia z VR, AR aj MR.

4. Špecifické formy adaptivity

- **Opticky adaptívne rozhrania:** Riešia ostrosť obrazu v závislosti od akomodácie oka (zaostrenia). **Varifokálne systémy** dynamicky menia ohniskovú vzdialenosť displeja alebo šošoviek podľa toho, kam sa používateľ pozerá, čím eliminujú VAC konflikt.
 - **Eyetracking (ETK):** Optické snímanie pohybu očí, ktoré umožňuje adaptívne ovládanie systému pohľadom a techniky ako *Foveated rendering* (vykresľovanie vysokého detailu len tam, kam oko skutočne hľadí).
 - **Bio-adaptované rozhrania:** Snímajú biologické funkcie používateľa (napr. svalové napätie cez EMG) a prispôsobujú reakciu systému jeho fyzickému stavu.
-

5) Technológia digitálneho dvojčaťa.

1. Definícia a pôvod digitálneho dvojčaťa

Digitálne dvojča (Digital Twin) je riešenie založené na zbere dát z reálnych (výrobných) procesov, ich vyhodnocovaní a následnom optimalizovaní s využitím technológií VR/XR.

- **História:** Koncept prvýkrát spomenul **David Gelernter** v roku 1991 vo svojej knihe *Mirror World*.
- **Priemyselné využitie:** Pre priemyselnú výrobu v súvislosti s **Industry 4.0** ho v roku 2002 rozpracoval **Michael Grieves**.
- **Podstata:** V princípe ide o sofistikovaného „avatara“ konkrétneho fyzického výrobku alebo procesu.

2. Úlohy a ciele digitálneho dvojčaťa

Hlavným cieľom je simulovať realitu v digitálnej forme, aby bolo možné rýchlo a exaktne odpovedať na dôležité otázky bez zásahu do skutočnej výroby. Medzi kľúčové úlohy patrí:

- **Simulácia a vizualizácia:** Prezentácia celkového 3D modelu výrobku alebo procesu z rôznych pohľadov.
- **Optimalizácia:** Usporiadanie zákaziek tak, aby výroba bola čo najrýchlejšia, najlacnejšia alebo najobjemnejšia.
- **Predikcia údržby:** Určenie presného času na výmenu nástrojov z dôvodu ukončenia ich životnosti.
- **Analýza porúch:** Zistenie príčiny nežiaducej udalosti a odporúčanie riešenia, ako jej predchádzať.

3. Prepojenie reality a virtuality (Slučka teleriadenia)

Digitálne dvojča vytvára uzavretý cyklus medzi fyzickým a virtuálnym svetom:

1. **Virtualizácia:** Proces prenosu údajov z reality (fyzického objektu) do virtuálneho prostredia prostredníctvom senzorov.
2. **Teleriadenie:** Spätná väzba, kedy rozhodnutia alebo optimalizácie vykonané vo virtuálnom svete ovplyvňujú reálny objekt. Pre tento proces je nevyhnutná kvalitná vizualizácia a čo najprirodzenejšie rozhranie.

4. Štruktúra modelu digitálneho dvojčaťa

Model digitálneho dvojčaťa sa skladá z dvoch hlavných častí:

- **Virtuálne prostredie:** Zahŕňa riadenie prostredia, dynamiku objektov, implementáciu fyziky a vizuálny model.
- **Virtuálny objekt:** Obsahuje riadenie (kinematika, parametre, skripty) a samotný model (geometrický, skeletálny, kĺbový).

5. Oblasti nasadenia

Okrem virtualizácie **výrobného procesu** v priemysle sa táto technológia využíva aj v:

- urbanistickom plánovaní a stavebníctve,
 - zdravotníctve,
 - automobilovom priemysle.
-

6) *Zmiešaná realita, funkcia systému ZR, klasifikácie ZR.*

1. Zmiešaná realita (MR – Mixed Reality)

Podľa Azumovej definície je **Zmiešaná realita** oblasť počítačového výskumu, ktorá sa zaoberá kombináciou reálneho sveta a počítačom generovaných dát (VR).

Základné charakteristiky MR:

- **Kombinuje** reálne a virtuálne prvky.
- Je **interaktívna** v reálnom čase.
- Je **registrovaná v 3D** (virtuálne objekty majú pevné miesto v priestore).

2. Milgramovo kontinuum (Realita – Virtualita)

Tento koncept (Paul Milgram a Fumio Kishino) definuje MR ako prechod medzi čisto fyzickou realitou a čisto virtuálnym prostredím.

- **Augmentovaná realita (AR):** Vkládanie virtuálnych objektov do reálneho sveta.
- **Augmentovaná virtualita (AV):** Vkládanie reálnych objektov do virtuálneho sveta.
- **Zmiešaná realita (MR):** Zahŕňa celé spektrum medzi týmito dvoma bodmi.

3. Funkcia a mechanizmy systému MR

Systém funguje na báze tzv. **XR rúry (pipeline)**, čo je životný cyklus spracovania dát od zberu požiadaviek až po vizualizáciu a spätnú väzbu.

Kľúčové technologické funkcie:

- **SLAM (Simultaneous Localization and Mapping):** Metóda, ktorá v reálnom čase simultánne určuje polohu zariadenia a vytvára mapu okolitého prostredia. Využíva dáta z kamier a inerciálne senzory (IMU).
- **3D registrácia:** Presné umiestnenie virtuálnych objektov v reálnom priestore tak, aby sa zachovali priestorové vzťahy a mierka.
- **Interakcia:** Virtuálne objekty dokážu reagovať na fyzické prostredie a používateľa (napr. gestá rúk).

4. Klasifikácie MR systémov

Systémy zmiešanej reality sa delia podľa viacerých kritérií:

A. Podľa kompozície vizuálneho vnemu:

- **MR s priamym pohľadom (Optical see-through):** Používateľ vidí realitu priamo cez priehľadné displeje, na ktoré sa premieta grafika (napr. MS HoloLens).
- **MR s nepriamym/sprostredkovaným pohľadom (Video see-through):** Realita je snímaná kamerami a následne zobrazená na nepriehľadnom displeji spolu s grafikou (napr. Meta Quest 3, Apple Vision Pro).

B. Podľa spôsobu synchronizácie (detekcie):

- **Systémy so značkami (Markered):** Využívajú exaktné fyzické značky (napr. QR kódy), ktoré systém rozpozná a nahradí objektom.
- **Systémy bez značiek (Markerless):** Využívajú rozpoznávanie tváre, obrazu, GPS, WiFi signál alebo 3D sken priestoru v reálnom čase (SLAM).

C. Podľa prostredia:

- **Interiérové** (vnútorné priestory).
- **Exteriérové** (vonkajšie prostredie).
- **Kombinované.**

D. Mannova klasifikácia (Medialita): Rozširuje Milgramovo kontinuum o os **mediality** (sprostredkovania). Rieši mieru prepojenia medzi svetmi, kde sú údaje transformované do "mediálnej" podoby (mediácia).

7) Typy ZR a možnosti ich využitia.

1. Typy MR podľa vizuálnej kompozície

Rozlišujeme dva základné technické prístupy k tomu, ako sa používateľovi spája realita s virtuálnymi prvkami:

- **MR s priamym pohľadom (Optical see-through):** Používateľ vidí reálny svet priamo cez polopriehľadné sklá (displeje), na ktoré sa premieta grafika.
 - *Príklad: Microsoft HoloLens (1, 2).*
- **MR s nepriamym / sprostredkovaným pohľadom (Video see-through):** Realita je snímaná kamerami a následne zobrazená na nepriehľadnom displeji v kombinácii s grafikou. Tento prístup je typický pre moderné VR headsety s funkciou "passthrough" alebo AR aplikácie v mobiloch.
 - *Príklad: Meta Quest 3, Apple Vision Pro.*

2. Typy MR podľa spôsobu synchronizácie v priestore

Tieto systémy určujú, ako virtuálny objekt "vie", kde v realite sa má zobrazíť:

- **Systémy so značkami (Markered):** Využívajú fyzické značky (napr. QR kódy alebo špeciálne obrazce) umiestnené v priestore. Keď kamera značku rozpozná, nahradí ju virtuálnym objektom.
- **Systémy bez značiek (Markerless):** Nepotrebuju vopred pripravené značky. Využívajú:
 - **Rozpoznávanie obrazu** (tváre, predmety).
 - **Geolokalizačné dáta** (GPS, WiFi signál).
 - **SLAM technológiu** (simultánne 3D skenovanie a mapovanie priestoru v reálnom čase).

3. Typy podľa prostredia

- **Interiérové:** Používané v budovách, laboratóriách alebo v priemyselných halách.
- **Exteriérové:** Určené na prácu vonku (využívajú najmä GPS).
- **Kombinované:** Umožňujú plynulý prechod medzi oboma prostrediami.

4. Možnosti využitia MR v praxi

Využitie MR prináša výhody najmä tam, kde je potrebné prepojiť digitálne informácie s fyzickou činnosťou:

- **Priemysel a architektúra:** Vizualizácia projektu pred realizáciou, odhaľovanie konštrukčných chýb v 3D modeli, orientácia v závode pomocou digitálnych pokynov.
- **Medicína:** Pomoc pri operáciách (zobrazenie orgánov priamo na pacientovi), terapeutické systémy (napr. pre paretických pacientov – tréning úchopu).
- **Výuka a tréning:** Školenie servisu zložitých strojov bez potreby ich rozoberania v realite, simulátory bezpečnosti (hasičský tréning, zváranie, ovládanie vysokozdvížných vozíkov).
- **Armáda a bezpečnosť:** Taktické zobrazenie informácií priamo v zornom poli vojaka.
- **Marketing a predaj:** Efektne prezentácie produktov, virtuálne katalógy a interaktívne kampane.
- **Hry a zábava:** Exteriérové hry založené na polohe alebo pokročilá domáca zábava.

8) Proces práce aplikácie ZR, implementácia značkovanej AR.

1. Proces práce aplikácie ZR (XR rúra)

Fungovanie aplikácie zmiešanej reality (ZR/MR) je postavené na tzv. **XR rúre (pipeline)**, čo je životný cyklus spracovania dát v reálnom čase. Tento proces integruje:

- **Logické jadro systému** (vyhodnocovanie stavu).
- **Dátové modely a assety** (virtuálne objekty).
- **Fyzikálne a simulačné moduly** (kolízie, správanie).

- **Renderovací a interakčný subsystém** (vizualizácia a spätná väzba).

2. Všeobecný algoritmus práce MR aplikácie

Bez ohľadu na typ detekcie (so značkami alebo bez) prebieha hlavný cyklus nasledovne:

1. **Snímanie:** Kamera neustále sníma reálnu scénu a posiela video do systému.
2. **Detekcia:** Program spracúva snímku po snímke a hľadá vopred definovanú predlohu alebo objekt.
3. **Výpočet:** Po rozpoznaní objektu sa vypočíta jeho **presná pozícia a orientácia** v priestore.
4. **Umiestnenie:** Virtuálny model sa vykreslí na vypočítanú pozíciu.
5. **Vizualizácia:** Používateľ vidí reálnu scénu rozšírenú o 3D model.

3. Implementácia značkovanej (marked) AR

Tento prístup využíva exaktné fyzické značky (napr. QR kódy alebo čiernobielye obrazce), ktoré slúžia ako kotvy pre virtuálny obsah.

A. Proces vytvárania scény (Príprava):

- **Zhromaždenie údajov a tvorba 3D modelu:** Export a kontrola modelu.
- **Definícia markera:** Vytvorenie grafickej značky a kontrola jej rozpoznateľnosti systémom.
- **Prepojenie:** Spojenie konkrétneho 3D modelu s danou značkou v prostredí vývojového nástroja.

B. Proces spracovania (Runtime):

Keď aplikácia beží, vykonáva tieto kroky:

1. **Hľadanie značky:** Video z kamery sa prevedie na **binárny (čiernobiely) obraz** pre ľahšiu identifikáciu.
2. **Identifikácia ID:** Systém zistí, o ktorú značku ide.
3. **Výpočet transformácie:** Určí sa 3D poloha a orientácia značky vzhľadom ku kamere.
4. **Zobrazenie:** Vykreslí sa priradený 3D objekt priamo na miesto značky v obraze.

4. Zosúladenie reálnej a virtuálnej scény

Kľúčom k stabilnej ZR je prepojenie súradnicových systémov:

- **Súradnicový systém reálnej scény:** Určený polohou markera v realite.
 - **Súradnicový systém kamery:** Mení sa podľa pohybu používateľa.
 - **Transformačná matica:** Matematický výpočet, ktorý zosúladí reálny pohľad s virtuálnym, aby objekt "neplával" a držal sa značky. Pre výpočet sa často určujú 4 vrcholy značky a normálové vektory rovín.
-

9) Metódy detekcie v ZR systémoch.

Tieto metódy sú kľúčové pre **markerless (bezznačkové)** systémy, pretože umožňujú systému "vidieť" a pochopiť reálne prostredie bez vopred pripravených grafických značiek.

Najviac používané metódy detekcie v MR

V súčasných systémoch sa primárne využívajú tri algoritmy:

1. **SIFT** (Scale Invariant Feature Transform)
2. **SURF** (Speeded-Up Robust Features)
3. **SLAM** (Simultaneous Localization and Mapping)

1. SIFT – Transformácia príznakov nezávislá od mierky

Tento algoritmus hľadá v obraze významné body, ktoré sú stabilné aj pri zmene mierky alebo otočení.

- **Proces práce SIFT:**
 1. **Vyhľadanie lokálnych extrémov:** Vytvorí sa viacero verzií obrázka s rôznym rozmazaním (Gaussovo rozmazanie) a hľadajú sa v nich extrémne body.
 2. **Spresenie bodov:** Zo získaných bodov sa vylúčia nestabilné a pri zvyšných sa presne určí ich poloha.
 3. **Priradenie orientácie:** Každému bodu sa priradí dominantný smer (orientácia), čím sa dosiahne **nezávislosť od rotácie**.
 4. **Vytvorenie deskriptorov:** Pre každý bod sa vypočíta číselný popis (deskriptor), ktorý slúži na hľadanie zhody medzi obrazmi.

2. SURF – Zrýchlené robustné príznaky

SURF je **novšia a rýchlejšia obdoba metódy SIFT**. Je navrhnutá tak, aby bola výpočtovo efektívnejšia pri zachovaní odolnosti voči rotácii a vzdialenosti kamery.

- **Kľúčové vlastnosti SURF:**
 - Využíva tzv. **integrálny obraz**, ktorý dramaticky zrýchľuje výpočet sumy hodnôt pixelov (stačia na to 4 rohové body).
 - Používa **determinant Hessianovej matice** na detekciu bodov záujmu.
 - Je ideálna pre rýchly popis obsahu obrazu a rekonštrukciu 2D/3D scén.

3. SLAM – Simultánna lokalizácia a mapovanie

SLAM je základom moderných MR systémov (napr. HoloLens) a mobilných AR aplikácií.

- **Funkcia:** Zariadenie v reálnom čase súčasne robí dve veci: **určuje svoju polohu** v priestore a zároveň **vytvára mapu** neznámeho prostredia.

- **Vstupy pre SLAM:**
 - **Obrazové dáta** z kamery.
 - **IMU senzory** (inerciálne meracie jednotky) – akcelerometer pre zrýchlenie, gyroskop pre rotáciu a magnetometer pre orientáciu k svetovým stranám.
- **Význam:** Umožňuje presnú registráciu virtuálnych objektov v priestore bez potreby akýchkoľvek značiek (markerless).

Technické limity detekcie

Pri implementácii týchto metód je potrebné počítať s obmedzeniami:

- **Výpočtová náročnosť:** Najmä mobilné zariadenia a bezdrôtové headsety potrebujú vysoký výkon pre spracovanie algoritmov v reálnom čase.
- **Vplyv prostredia:** Detekcia môže zlyhať pri slabom osvetlení, na lesklých/reflexných povrchoch alebo pri príliš homogénnych (jednoliatych) stenách bez textúry.
- **Latencia:** Oneskorenie medzi snímaním a vykreslením môže narušiť stabilitu virtuálnych objektov (tzv. "plávanie" objektov).

10) Limity systémov ZR.

1. Technické limity MR systémov

Tieto limity priamo ovplyvňujú stabilitu a uveriteľnosť vkladania virtuálnych objektov do reality:

- **Výpočtová náročnosť:** Systémy vyžadujú vysoký výkon pre spracovanie algoritmov v reálnom čase, čo je kritické najmä pri mobilných zariadeniach a bezdrôtových headsetoch.
- **Latencia (oneskorenie):** Časový rozdiel medzi snímaním reality a vykreslením virtuálneho objektu. V systémoch pre viacerých používateľov sa pridáva aj sieťová latencia (synchronizácia dát).
- **Presnosť registrácie v priestore:** Problém s "plávaním" objektov. Dochádza k vychýleniu polohy v čase a kumulácii chýb merania senzorov.
- **Vplyv prostredia:**
 - **Osvetlenie:** Detekcia zlyháva pri príliš slabom alebo veľmi ostrom svetle.
 - **Povrchy:** Problémom sú lesklé, reflexné (zrkadlové) alebo príliš homogénne povrchy (napr. biela stena bez textúry).
 - **Zašumené pozadie:** Príliš členité prostredie môže zmiatť algoritmy rozpoznávania.

2. Obmedzenia vizualizačného podsystému

Okrem grafického výkonu naráža MR na hardvérové bariéry zariadení:

- **Energetická a tepelná náročnosť:** Spotreba energie a prehrievanie procesorov vo vnútri HMD okuliarov.

- **Pamäťový systém:** Limity v priepustnosti pamäte a rýchlosti zberníc.
- **Limity vykresľovania:** Rýchlosť zaplňania GPU (fill-rate) a zložitosť shaderov.

3. Biologické limity a vizuálne konflikty

Najväčšou bariérou pre dlhodobé používanie ZR sú reakcie ľudského tela:

- **VAC konflikt (Vergence-Accommodation Conflict):** Ide o nesúlad medzi konvergenciou očí (uhol pohľadu na objekt) a akomodáciou (zaostrenie šošovky). V okuliaroch sú oči zaostrené na fixnú rovinu displeja, ale mozog sa snaží zaostriť na virtuálny objekt v inej hĺbke. Spôsobuje to únavu očí, bolesti hlavy a nevoľnosť.
- **Real-time obmedzenia:** Systém musí udržať stabilnú obnovovaciu frekvenciu (60–120 Hz). Ak klesne pod tento limit, dochádza k strate imerzie a nastupuje **kybernetická nevoľnosť (cybersickness)**.

4. Kompromisy pri návrhu (Výzvy)

Dizajnéri ZR systémov musia neustále hľadať rovnováhu medzi protichodnými faktormi:

- **Univerzálnosť vs. Optimalizácia** pre konkrétny hardvér.
- **Kvalita obrazu vs. Latencia** (čím krajší obraz, tým dlhšie sa renderuje).
- **Realizmus vs. Výkon** (imerzia verzus plynulosť systému).

11) Kolaboratívna XR/VR, definícia, vlastnosti, klasifikácia, typy, využitie.

1. Definícia Kolaboratívnej VR/XR

- **Systém Kolaboratívnej VR/XR (CVR/CXR):** Je systém založený na báze technológií virtuálnej alebo rozšírenej reality, ktorého účelom je sprostredkovať **spoločne zdieľané virtuálne prostredie** medzi viacerými používateľmi v reálnom čase.
- **Kolaboratívne Virtuálne Prostredie (CVE):** Predstavuje samotný virtuálny 3D priestor, ktorý poskytuje platformu pre viacpoužívateľskú komunikáciu a interakciu.

2. Kolaborácia vs. Kooperácia

V kontexte XR systémov sa tieto pojmy často používajú ako synonymá, no majú rozdielny význam:

- **Kolaborácia (tesne-viazaná):** Aktéri pracujú spolu, aktívne sa ovplyvňujú, tvoria a riešia problémy spoločne so zdieľanou zodpovednosťou.
- **Kooperácia (voľne-viazaná):** Aktéri pracujú na spoločnom ciele, ale každý rieši svoju časť úlohy samostatne (paralelne). Výsledok vzniká „poskladaním“ častí.

3. Vlastnosti kolaboratívneho prostredia (CVE)

- **Integrácia skupiny:** Podpora aktivít viacerých používateľov naraz.
- **Zdieľanie:** Spoločné vnímanie virtuálneho, zmiešaného alebo zlúčeného priestoru.
- **Spoločný zámer:** Smerovanie k dosiahnutiu spoločného cieľového riešenia.

- **Multimodalita:** Kombinácia viacerých typov vstupov a výstupov v reálnom čase.
- **Interaktivita:** Možnosť komunikovať a operovať v zdieľanom prostredí.

4. Klasifikácia systémov CVR/CXR

Systémy sa delia podľa štyroch základných aspektov:

1. Podľa účelu využitia:

- **Viacúčelové:** Všeobecné použitie, široké spektrum úloh (napr. sociálne siete vo VR).
- **Špecifické:** Riešenie konkrétnych zložitých úloh (napr. simulátor operácie).

2. Podľa dostupnosti: Lokálne vs. Globálne systémy.

3. Podľa architektúry: Klient-server, Peer-to-peer alebo Distribuovaná serverová architektúra.

4. Podľa modelov konzistencie: Centralizované, Aktívne replikované alebo Distribuované virtuálne prostredia.

5. Typy podľa dostupnosti (Lokálne vs. Globálne)

- **Lokálne zdieľané systémy:** Kolaborácia prebieha v rámci **spoločného fyzického priestoru**. Umožňujú prirodzenú komunikáciu (používatelia sú v jednej miestnosti). Príkladom je systém CAVE alebo lokálne prepojené headsety.
- **Globálne zdieľané systémy:** Umožňujú **vzdialenú kolaboráciu** rozsiahlej skupiny používateľov cez sieťovú infraštruktúru (internet). Príkladom je systém **LIRKIS G-CVE**, ktorý poskytuje prostredie cez webové prehliadače.

6. Aplikačné nasadenie a využitie

Kolaboratívna XR nachádza uplatnenie v mnohých oblastiach:

- **Vzdelávanie a tréning:** Virtuálna výuka, tréningové aplikácie pre skupiny.
- **Priemysel a konštrukcia:** Dizajnové a konštrukčné aplikácie, kde inžinieri spoločne pracujú na 3D modeli.
- **Medicína:** Skupinové medicínske simulácie alebo konzultácie nad 3D modelmi orgánov.
- **Simulácie:** Komplexné systémy (napr. riadenie letovej prevádzky alebo vojenské simulácie).
- **Zábava a kultúra:** Herné aktivity a virtuálne prehliadky múzeí či galérií pre skupiny.

12) Lokálne a globálne zdieľané systémy.

1. Rozdelenie podľa dostupnosti (Dostupnosť CVR/CXR)

Systémy kolaboratívnej virtuálnej/rozšírenej reality sa podľa miery geografického rozptýlenia používateľov delia na:

- **Lokálne zdieľané systémy** a virtuálne prostredia.
- **Globálne zdieľané systémy** a virtuálne prostredia.

2. Lokálne zdieľané systémy

Tieto systémy sú určené pre používateľov, ktorí sa nachádzajú v **spoločnom fyzickom priestore** (napr. v jednej miestnosti alebo laboratóriu).

- **Vlastnosti:**

- Umožňujú **prirodzenú komunikáciu** používateľov tvárou v tvár.
- Sú limitované z hľadiska vzdialenej kolaborácie (používateľ "zvonku" sa spravidla nemôže pripojiť).

- **Typy používateľského prístupu v lokálnych systémoch:**

1. **Centralizovaný prístup:** Skupina používateľov pristupuje k jednému centrálnemu výpočtovému systému, ktorý generuje zdieľané virtuálne prostredie (napr. systém CAVE).
2. **Oddelený prístup:** Každý používateľ má vlastný výpočtový systém (napr. vlastný headset), ale všetky sú prepojené cez lokálny komunikačný uzol v rámci jednej miestnosti.

3. Globálne zdieľané systémy

Umožňujú spoluprácu používateľov bez ohľadu na ich fyzickú polohu, čím vytvárajú **globálne zdieľaný priestor**.

- **Vlastnosti:**

- Sprostredkujú **vzdialenú kolaboráciu**.
- Sú navrhnuté pre **rozsiahle skupiny** používateľov.
- Pre ich fungovanie je **nevyhnutná sieťová infraštruktúra** (internet) a výpočtový serverový uzol.

4. Riadenie vizualizácie v zdieľaných systémoch

Spôsob, akým používatelia vidia virtuálny svet, závisí od architektúry systému:

- **Vizualizácia riadená jedným používateľom:** Typická pre lokálne centralizované systémy (napr. CAVE), kde jeden hlavný používateľ ovláda pohľad kamery pre všetkých ostatných.
- **Vizualizácia riadená každým používateľom:** Každý účastník má vlastnú nezávislú kameru a pohľad do scény. Tento model sa využíva v globálnych systémoch a lokálnych systémoch s oddeleným prístupom (typicky pri použití HMD/okuliarov).

5. Príklad: Systém LIRKIS G-CVE

LIRKIS G-CVE (Global Collaborative Virtual Environment) je príkladom moderného globálneho systému vyvinutého na FEI TU v Košiciach.

- **Účel:** Poskytovať globálne prístupné prostredia cez **webové prehliadače**.

- **Architektúra:** Klient-server založená na protokole **WebSocket**.
 - **Technológie:** Využíva JavaScriptové rámce ako **Node.js** (paralelné pripojenie), **Express.js** (API servera) a **Networked-Aframe** (synchronizácia 3D objektov a interakcií).
-

13) Konceptie aplikačnej architektúry kolaboratívnej XR/VR.

1. Základné architektonické vzory

Výber architektúry ovplyvňuje výkon systému, jeho stabilitu a schopnosť synchronizovať viacerých používateľov. Rozlišujeme tri základné konceptie:

- **Vzor Klient-server:** Najčastejšie používaný model s centrálnym riadením.
- **Vzor Peer-to-peer (P2P):** Decentrálny model, kde sú si všetky uzly rovné.
- **Distribovaná serverová architektúra:** Pokročilý model využívajúci viacero špecializovaných serverov.

2. Rozhodovacia autorita

Kľúčovým prvkom architektúry je určenie **authority**, teda elementu, ktorý rozhoduje o tom, aký je momentálne platný stav virtuálneho sveta.

- **Autoritatívny server:** O platnom stave rozhoduje len server (vysoká konzistencia).
- **Autoritatívny klient:** Rozhodnutia sa robia na strane používateľa (rýchlejšia odozva, riziko nekonzistencie).
- **Hybridná autorita:** Kombinácia oboch prístupov.
- **Význam:** Nesprávne určená autorita môže viesť k **nekonzistencii prostredia** (rozpor v tom, čo vidia rôzni používatelia) a bezpečnostným rizikám.

3. Detailný pohľad na architektúry

- **Klient-server:**
 - **Server** slúži na centralizáciu a synchronizáciu dát, spravuje prístupy a odľahčuje výpočtový výkon klientskych zariadení.
 - **Klient** vykonáva obojstrannú komunikáciu a zobrazuje interakcie ostatných členov skupiny.
- **Distribovaná serverová architektúra:**
 - Využíva najmenej dva serverové uzly.
 - **Primárny server** rieši prístupy klientov a sprostredkúva prostredie.
 - **Sekundárny server** sa stará o špecifické podsystémy (fyzika, komplexné dáta).

4. Plný distribuovaný systém XR

Tento koncept implementuje prostredie ako súbor modulov prepojených cez **komunikačný uzol**. Medzi základné moduly patria:

- **Modul používateľa:** Rieši interakcie so scénou.
- **Úložisko scény:** Obsahuje dáta o všetkých objektoch.
- **Synchronizačný modul:** Riadi rovnomerné **plynutie virtuálneho času**.
- **Fyzikálny a pohybový modul:** Implementuje fyzikálne zákony a aktualizuje polohy objektov.

5. Príklad z praxe: LIRKIS G-CVE

Systém vyvinutý na KPI FEI TU v Košiciach využíva **architektúru klient-server** prispôsobenú pre webové prostredie:

- **Webový server** riadi komunikáciu medzi web-klientmi cez protokol **WebSocket**.
- Využíva rámce **Node.js** (paralelné pripojenia) a **Express.js** (API servera).
- Nástroj **Networked-Aframe** zabezpečuje, že sa medzi klientmi v reálnom čase synchronizujú súradnice avatarov, interakcie a vlastníctvo objektov.

14) Zdieľané virtuálne prostredia, ich konzistencia a modely.

1. Zdieľané virtuálne prostredie (CVE)

Kolaboratívne (zdieľané) virtuálne prostredie (CVE) je virtuálny trojrozmerný priestor, ktorý umožňuje viacerým používateľom súčasne komunikovať, interagovať a zdieľať spoločné vnímanie digitálneho sveta v reálnom čase.

2. Konzistencia v zdieľaných prostrediach

Konzistencia je stav, kedy **všetci používatelia vidia v danom čase to isté**.

- **Problém:** Keď používatelia vykonávajú aktivity, modifikujú zdieľaný obsah.
- **Požiadavka:** Každá zmena musí byť okamžitá a bez straty sprostredkovaná všetkým ostatným účastníkom.
- **Rola authority:** Konzistencia priamo závisí od určenia **rozhodovacej authority** (server alebo klient), ktorá určí, ktorý stav objektu je ten platný.

3. Modely konzistencie zdieľaných prostredí

Podľa tvojich podkladov rozlišujeme štyri základné modely zabezpečenia konzistencie:

A. Centralizované virtuálne prostredie

Všetky dáta a zmeny spravuje jeden centrálny bod (server).

- **Pravidlá:**

1. Pri pripojení dostane používateľ kompletnú kópiu (replikáciu) aktuálneho stavu prostredia.
2. Modifikácie môžu vykonávať všetci súčasne.
3. Po zmene sa aktualizuje centrálny stav a následne sa odošle (replikuje) všetkým používateľom.
4. **Stav sa uchováva** aj po ukončení kolaborácie pre budúce použitie.

B. Aktívne replikované virtuálne prostredie

Neexistuje centrálny server, dáta sa prenášajú priamo medzi uzlami (P2P štýl).

- **Pravidlá:**

1. Každý používateľ musí mať prostredie vopred nainštalované v rovnakom počiatočnom stave.
2. Komunikačný uzol len preposiela dáta, ale ich necentralizuje.
3. Novopripojenému používateľovi posielajú aktuálny stav ostatní účastníci.
4. Zmeny sa replikujú hromadne všetkým uzlom.
5. **Stav sa neuchováva** po ukončení kolaborácie.

C. Distribuované virtuálne prostredie

Tento model využíva kombináciu viacerých serverov (napr. primárny pre prístupy a sekundárny pre fyziku). Z hľadiska konzistencie preň platia **rovnaké pravidlá ako pre centralizovaný model**.

D. Zdieľanie vlastníctva údajov (Ownership)

Tento model rieši konzistenciu cez to, kto má práve právo s objektom hýbať. Obsahuje tri kľúčové udalosti:

- **Získanie vlastníctva:** Používateľ „uchopí“ objekt a systém mu pridelí právo na interakciu.
- **Odovzdanie vlastníctva:** Používateľ cielene odovzdá objekt inému kolegovi.
- **Odstúpenie vlastníctva:** Používateľ sa vzdá kontroly a ponechá objekt voľne v prostredí.

15) Vlastníctvo údajov a virtuálnych objektov v systémoch zdieľanej XR/VR, mechaanizmy replikácie údajov, objektov a scén.

1. Vlastníctvo údajov a virtuálnych objektov (Ownership)

V zdieľaných systémoch XR/VR je **vlastníctvo (ownership)** kľúčovým mechanizmom pre udržanie konzistencie. Určuje, ktorý používateľ má v danom momente právo manipulovať s konkrétnym objektom, čím sa predchádza konfliktom pri súbežnej interakcii viacerých osôb.

Zdieľanie vlastníctva obsahuje tri základné udalosti:

- **Získanie vlastníctva:** Používateľ získa prístup k virtuálnemu objektu a začne s ním interagovať (napr. ho uchopí).
- **Odobranie vlastníctva:** Používateľ cielene predá alebo odovzdá objekt inému účastníkovi kolaborácie.
- **Odstúpenie vlastníctva:** Používateľ sa vzdá kontroly nad objektom a ponechá ho voľne v prostredí pre ostatných.

2. Mechanizmy replikácie údajov

Replikácia je proces šírenia zmien o stave objektov a prostredia medzi všetkých pripojených používateľov. V systémoch XR rozlišujeme dva základné typy replikácie:

A. Synchronná replikácia

Používa sa na kritické systémové požiadavky, ktoré **musia byť spracované** predtým, než systém dovolí pokračovať v danej aktivite.

- **Charakteristika:** Žiadateľ (používateľ) musí čakať na spätnú odpoveď od systému (potvrdenie). Ostatní používatelia však môžu počas tohto spracovávania pokračovať vo svojej činnosti.
- **Príklady:** Pripojenie alebo odpojenie používateľa zo systému, zmena identity (avatara).

B. Asynchronná replikácia

Používa sa pre aktivity, ktoré prebiehajú súbežne a vyžadujú vysokú frekvenciu aktualizácií bez nutnosti explicitného potvrdzovania každého kroku serverom.

- **Charakteristika:** Dáta sú replikované v krátkych, opakujúcich sa časových intervaloch (v slučke) bez priameho spracovávania požiadaviek systémom.
- **Príklady:** Pohyb používateľa (kamery/avatara) v priestore, zasielanie komunikačných správ v chate.

3. Replikácia objektov a scén podľa modelu konzistencie

Spôsob, akým sa replikujú celé scény, závisí od zvolenej architektúry:

- **Centralizovaná replikácia:** Po pripojení používateľ obdrží kompletnú kópiu (replikáciu) celého prostredia. Každá modifikácia sa najprv odošle na server a ten ju následne replikuje všetkým ostatným.
- **Aktívna replikácia (P2P):** Komunikačný uzol údaje necentralizuje, prebieha len priama výmena dát medzi účastníkmi. Ak sa pripojí nový člen, aktuálny stav scény mu replikujú ostatní používatelia.

4. Praktický príklad: LIRKIS G-CVE

V systéme **LIRKIS G-CVE** sa prostredníctvom komponentov (napr. Networked-Aframe) a protokolu WebSocket v reálnom čase replikujú:

- vlastníctvo objektov,
 - vlastnosti a súradnice avatarov,
 - interakcie používateľov.
-

16) Latencia v systémoch zdieľanej XR/VR.

1. Definícia latencie v XR

Latencia predstavuje časové oneskorenie medzi vstupnou aktivitou používateľa (napr. pohybom hlavy alebo stlačením tlačidla) a odozvou systému, ktorú používateľ vníma. V systémoch zdieľanej reality (CVR/CXR) sa pod **sieťovou latenciou** rozumie časové oneskorenie medzi vykonaním interakcie jedným používateľom a jej zobrazením všetkým ostatným účastníkom.

2. Špecifiká zdieľaných (multi-user) systémov

V systémoch pre jedného používateľa je latencia daná najmä výkonom lokálneho hardvéru. V **multi-user systémoch** je však problém latencie oveľa výraznejší, pretože k procesu snímania a vykresľovania pribúda nevyhnutná **sieťová komunikácia a synchronizácia** dát medzi účastníkmi.

3. Zdroje latencie v zdieľanej realite

Podľa zdrojov sú hlavnými príčinami oneskorenia v kolaboratívnom prostredí:

- **Sieťová infraštruktúra:** Technické parametre siete a fyzický prenos dát.
- **Spracovanie dát serverom:** Čas potrebný na to, aby server prijal požiadavku, vyhodnotil logiku a aktualizoval stav prostredia.
- **Mechанизmus replikácie:** Proces šírenia zmien k ostatným klientom.
- **Vykresľovanie na klientoch:** Čas potrebný na renderovanie prijatých zmien na zariadeniach ostatných používateľov.

4. Typy latencií v XR systémoch

Pri implementácii je potrebné riešiť štyri základné roviny latencie:

- **Vstupná latencia:** Oneskorenie pri snímaní pohybu, gest alebo bio-senzorických vstupov.
- **Procesná (výpočtová) latencia:** Oneskorenie v rámci hlavnej procesnej slučky jadra systému.
- **Sieťová latencia:** Riešená infraštruktúrne (kvalita siete) alebo softvérovým manažmentom (vhodná architektúra, napr. klient-server).
- **Vizualizačná latencia:** Súvisí so spôsobom výpočtu obrazu, renderovaním a technikami ako stereoskopia. Špecifickým meradlom je **motion-to-photon latencia**, čo je interval medzi pohybom hlavy a zmenou pixelov na displeji.

5. Vplyv latencie na používateľa a systém

Vysoká latencia má v zdieľanom prostredí tieto negatívne dôsledky:

- **Narušenie konzistencie:** Používatelia môžu vidieť stav objektov rozdielne v rovnakom čase.
- **Problémy so synchronizáciou:** Pohyby avatarov a interakcie viacerých osôb nie sú zosúladené.
- **Zníženie prezencie:** Používatelia strácajú pocit spoločnej prítomnosti v priestore.
- **Zdravotné riziká:** Oneskorenie obrazu za pohybom vedie k **kybernetickej nevoľnosti (cybersickness)** a strate imerzie.

17) Interakcia používateľov v systémoch zdieľanej XR/VR a viacpoužívateľský prístup, mechanizmy riadenia, bezpečnosť viacpoužívateľského prístupu.

1. Interakcia v zdieľanej XR/VR

V zdieľanom prostredí (**CVE – Collaborative Virtual Environment**) prebieha interakcia v reálnom čase medzi viacerými používateľmi a objektmi.

- **Základné techniky:** Selekcia (označenie objektu), manipulácia (hýbanie s ním) a lokomócia (pohyb používateľa v scéne).
- **Multimodalita:** Kombinácia viacerých typov vstupov a výstupov (obraz, zvuk, hmat) pre lepší zážitok.
- **Kolaborácia vs. Kooperácia:** Kolaborácia je tesne-viazaná spoločná práca so zdieľanou zodpovednosťou, zatiaľ čo kooperácia je voľne-viazaná paralelná práca na častiach úlohy.

2. Mechanizmy riadenia interakcie

Keď viacerí používatelia chcú naraz hýbať s tým istým objektom, vznikajú konflikty. Systémy to riešia dvoma spôsobmi:

- **Mechanizmus s blokovaním:** Synchronný prístup, kde objekt môže v danom čase ovládať **len jeden používateľ**.
 - *Algoritmus:* Používateľ požiada o prístup -> systém skontroluje stav (ak je voľný, uzamkne ho pre neho) -> po skončení interakcie sa objekt odomkne pre ostatných.
- **Mechanizmus s voľnou interakciou:** Používatelia si môžu pri ovládaní objektu **konkurovať**.
 - *Algoritmus:* Ak začne interagovať druhý používateľ, systém automaticky ukončí interakciu toho prvého a prideli kontrolu novému žiadateľovi.

3. Viacpoužívateľský prístup a jeho riadenie

Prístup do systému je riadený na dvoch úrovniach:

- **Riadenie podľa identifikácie:** Proces overenia identity používateľa.
 - *Postup:* Zaslanie údajov -> Prijatie -> **Identifikácia** (kto to je) -> **Autorizácia** (čo môže robiť) -> Povolenie/zamietnutie prístupu.

- **Riadenie podľa hierarchie (RBAC):** Využíva model **Role-based Access Control** (prístup na základe rolí).
 - Používateľom sú pridelené kompetencie podľa ich roly (napr. **Editor** môže meniť obsah, **Inštruktor** riadiť kontext, **Pozorovateľ** len sledovať).

4. Bezpečnosť viacpoužívateľského prístupu

Bezpečnosť v systémoch zdieľanej reality zahŕňa technické aj etické aspekty:

- **Rozhodovacia autorita:** Je kľúčové určiť, či o stave sveta rozhoduje server (bezpečnejšie) alebo klient (rýchlejšie). Nesprávne určená autorita vedie k nekonzistencii a bezpečnostným dieram.
- **Ochrana súkromia:** Systémy zbierajú citlivé dáta (pohyby tela, hlas, biometria, emočné reakcie), ktoré musia byť chránené pred zneužitím.
- **Vlastníctvo dát:** Rieši sa otázka, komu patria virtuálne objekty a ako sú dáta ukladané v cloude.
- **Fyzická bezpečnosť:** Potreba definovať ochranné zóny, aby používatelia v zápale interakcie nenarazili do prekážok v reálnom svete.

18) Základné implementačné stupne XR systému.

1. Implementačné stupne z pohľadu IT

Z hľadiska informačných technológií sa implementácia XR systému delí na tri základné roviny:

- **Technické prostriedky:** Hardvérová a konštrukčná časť systému.
- **Programové prostriedky:** Softvérové jadro, operačný systém a nástroje pre vývoj.
- **Ostatné prostriedky:** Nevyhnutná podpora z iných vedeckých odborov a bezpečnostné pravidlá.

Dobre navrhnutý systém je výsledkom rovnováhy medzi týmito prostriedkami a rešpektovaním ľudských limitov.

2. Technické prostriedky (Hardvér)

Zahŕňajú fyzickú stavbu a infraštruktúru systému:

- **Konštrukcia:** Mechanická a elektrická (silová aj nízkonapäťová) časť.
- **Podsystemy:** Koncepcia vizualizačného (najdôležitejší), akustického, kinematického a hmatového podsystemu.
- **Senzorické a sieťové systémy:** Snímače polohy a infraštruktúra pre prenos dát.
- **Výpočtový systém:** Architektúra (jedno/viacjadrová, paralelná, distribuovaná) a typy úložísk (lokálne/sieťové).

3. Programové prostriedky (Softvér)

Zabezpečujú logiku a fungovanie virtuálneho sveta:

- **Formalizácia (Definičné prostriedky):** Definovanie statických a dynamických objektov, scén a prírodných javov.
- **Jadro systému:** Riešenie matematických transformácií a detekcia kolízií.
- **Opis dát:** Programové prostriedky pre opis dátových štruktúr a modelov (napr. assety).
- **Skriptovanie:** Nástroje pre definovanie správania a interakcií v reálnom čase.

4. Ostatné prostriedky (Interdisciplinárna podpora)

Keďže XR systémy úzko interagujú s človekom, vyžadujú podporu z viacerých oblastí:

- **Vedecká podpora:** Fyzikálna, medicínska, neurologická a psychologická podpora pre zabezpečenie komfortu.
- **Bezpečnosť a hygiena:** Hygienické pravidlá (najmä pri zdieľaní hardvéru) a expertízna podpora pre špecifické nasadenia.

5. Procesný pohľad na implementáciu

Implementácia prebieha ako viac-vrstvový proces založený na tzv. **virtualizačnom reťazci**:

1. Zber a analýza požiadaviek.
2. Formalizácia systému (popis logického jadra).
3. Modelovanie virtuálneho priestoru a objektov (tvorba assetov).
4. Simulácia procesov a interakcií.
5. Vizualizácia a spätná väzba používateľa.

6. Bázický implementačný reťazec (Model A-E-R-H)

V súčasnej praxi sa implementácia XR systému často delí na štyri nadväzujúce vrstvy:

- **A (Assetová vrstva):** Tvorba obsahu (napr. v Blenderi).
 - **E (Engine):** Správa scény a rendering (napr. Unity, Three.js).
 - **R (Runtime):** Podporné knižnice a API (napr. OpenXR, WebXR).
 - **H (Hardvér):** Cieľové zariadenie (napr. Meta Quest).
-

19) Meranie komfortu používateľa XR systémov.

1. Teoretický základ a princípy

Meranie komfortu v XR systémoch vychádza z **ergonómie interakcie človek-systém** (HCI – Human Computer Interface) a dizajnu zameraného na človeka (HCD) podľa noriem **ISO 9241-210** a **ISO 9241-11:2018**. Kľúčovým poznatkom je, že **vyššia technologická úroveň systému neznamena automaticky vyšší komfort** používateľa. V súčasnosti neexistuje celosvetovo prijatá jednotná metodika, preto sa využívajú rôzne kombinácie prístupov.

2. Subjektívne (postojové) metódy merania

Tieto metódy sú založené na individuálnych hodnoteniach účastníkov, ktoré sa získavajú prostredníctvom dotazníkov, rozhovorov alebo videozáznamov. Medzi najpoužívanéjšie štandardizované nástroje patria:

- **SUS (System Usability Scale):** Škála na meranie celkovej použiteľnosti systému.
- **NASA-TLX (Task Load Index):** Viacrozmerný index na hodnotenie kognitívnej a mentálnej záťaže pri plnení úloh.
- **VRSQ (Virtual Reality Sickness Questionnaire):** Špecifický dotazník zameraný na symptómy kybernetickej nevoľnosti.
- **IPQ (Igroup Presence Questionnaire):** Nástroj na meranie miery prezencie (pocitu prítomnosti) vo virtuálnom prostredí.

3. Objektívne metódy merania

Sú založené na **kvantitatívnom meraní** pomocou senzorov, ktoré monitorujú fyziologické alebo behaviorálne reakcie používateľa v reálnom čase. Zameriavajú sa najmä na dva parametre:

- **Efektívnosť / Výkonnosť (Efficiency):** Meria čas potrebný na splnenie úlohy a počet pokusov s cieľom dosiahnuť výsledok s čo najmenším úsilím.
- **Efektivita / Účinnosť (Effectiveness):** Hodnotí mieru úspešnosti používateľa, správne poradie vykonaných krokov a počet chýb či neúspešných pokusov.

4. Ďalšie sledované parametre komfortu (UX)

Okrem vyššie uvedených metód sa pri vyhodnocovaní používateľského zážitku (UX) sledujú aj nasledujúce faktory:

- **Charakteristika priestoru:** Rozmer virtuálneho priestoru (2D vs. 3D).
 - **Fyzikálne veličiny:** Miera simulovanej vzdialenosti, objemu, jas, farieb alebo zvuku.
 - **Interakcia a biometria:** Počet interakcií s objektmi a meranie aktivít konkrétnych častí tela (hlava, prsty, oči).
 - **Prispôsobenie a hardvér:** Adaptácia pre pravákov/ľavákov a počet či **multimodalita V/V zariadení** (kombinácia myši, headsetu, senzorov a pod.).
-

20) Koncepcia distribuovaného systému XR.

1. Základná koncepcia distribuovanej architektúry

Distribuovaná serverová architektúra je postavená na báze modelu klient-server, ale na rozdiel od neho využíva **najmenej dva serverové uzly**. Tento prístup umožňuje rozdeliť výpočtovú záťaž a špecializovať jednotlivé časti systému.

- **Primárny server:** Zodpovedá za spracovanie klientskych prístupov a sprostredkovanie centralizovaného virtuálneho prostredia (centralizácia dát kolaborácie).
- **Sekundárny server:** Slúži na spracovávanie špecifických podsystémov a komplexných dát (distribúcia podsystémov).

2. Konzistencia v distribuovanom systéme

Z hľadiska udržiavania jednotného stavu prostredia platia pre distribuované virtuálne prostredie **rovnaké pravidlá ako pre centralizovaný model**. To znamená, že po pripojení používateľ dostane kompletnú replikáciu scény a každá modifikácia sa musí spätne replikovať všetkým účastníkom cez server.

3. Modulárna štruktúra plného distribuovaného systému

Moderný distribuovaný systém XR sa v rámci implementácie delí na špecializované moduly prepojené cez **komunikačný uzol**:

- **Modul používateľa:** Zabezpečuje interakciu so scénou a prijíma aktualizované údaje o stave sveta.
- **Úložisko virtuálnej scény:** Uchováva dáta o všetkých objektoch v prostredí, generuje notifikácie o zmenách a prijíma aktualizácie.
- **Synchronizačný modul:** Kľúčový modul, ktorý riadi **plynutie virtuálneho času** a vysiela synchronizačné signály a udalosti.
- **Fyzikálny a pohybový modul:** Implementuje fyzikálne zákony, deteguje kolízie a aktualizuje polohy a stavy objektov.

4. Príklady implementačných modulov

V praxi sa distribuovaný systém skladá z nasledujúcich funkčných častí:

- Komunikačný modul a úložisko údajov.
- Modul vizualizácie a zachytenia/prehrávania zvuku.
- Moduly pre spracovanie vstupov (údaje z polohových senzorov, gestá).
- Moduly pre logiku (skriptované animácie, manipulácia s objektmi, detekcia kolízií).

5. Význam distribuovanej koncepcie

Hlavným prínosom je **škálovateľnosť** a schopnosť spracovávať náročné výpočty (napr. fyziku alebo renderovanie) oddelene od správy sieťovej komunikácie, čo znižuje celkovú latenciu a zvyšuje stabilitu systému pre veľký počet používateľov.

21) Základný návrh XR systému na interakčnej báze, jeho rozhranie a bázičné koncepcie.

1. Návrh XR systému na interakčnej báze

Základný návrh XR systému vychádza zo **systému interakcie**, ktorý je v podkladoch označovaný akronymom **DEDSHO**. Interakcia nie je lineárna, ale prebieha v uzavretej slučke (feedback loop) medzi používateľom a strojom:

1. **Používateľ** vykonáva aktivitu cez rozhranie.
2. **Vstupný podsystem** zachytáva dáta (gestá, polohu, biometriu).
3. **Jadro systému** vyhodnocuje logiku, transformácie a kolízie v reálnom čase.
4. **Výstupný podsystem** generuje vnemy (obraz, zvuk, haptiku).
5. **Rozhranie** sprostredkúva tieto vnemy používateľovi, čím sa uzatvára slučka spätnej väzby.

Tento proces spracovania dát v reálnom čase sa nazýva **XR rúra (pipeline)** a integruje logické jadro, assety, fyzikálne moduly a renderovací subsystém.

2. Rozhrania XR systémov

Rozhranie (**Interface**) je kľúčový prvok, ktorý oddeľuje a zároveň spája digitálny svet s používateľom. Podľa technologickej úrovne rozlišujeme štyri typy rozhraní:

- **Priame, exaktné a mechanické:** Fyzické ovládacie prvky (napr. ovládacie panely).
- **Znakové:** Textové príkazy (CLI, terminál).
- **Grafické:** Štandardné vizuálne okná a ikony (GUI).
- **Bio-adaptované a prirodzené:** Najmodernejšia kategória pre XR systémy.

Prirodzené vs. Bio-adaptované rozhrania

- **Prirodzené (naturálne):** Sú prispôsobené človeku tak, aby systém ovládal čo najkomfortnejšie, typicky pomocou **gest alebo hlasu**.
- **Bio-adaptované:** Snímajú biologické funkcie (vstup), ako je **EEG (mozgová aktivita)**, **EMG (svalové napätie)**, tep alebo teplota, a na ich základe adaptujú reakciu systému.

Architektonická väzba rozhraní

XR rozhrania sa snažia o **posilnenie pohltenia (imerzie)** tým, že využívajú multimodálnosť – teda kombináciu vizuálnych, sluchových a hmatových vnemov tak, aby boli čo najbližšie reálnym ľudským vnemom.

3. Bázické koncepcie návrhu (Implementačný reťazec)

Pri návrhu a implementácii sa využíva tzv. **virtualizačný reťazec**, ktorý pozostáva z krokov: zber požiadaviek, formalizácia (popis jadra), modelovanie objektov (assets), simulácia a vizualizácia.

V súčasnej praxi sa presadil **bázický implementačný reťazec model A-E-R-H**:

- **A (Assetová vrstva):** Tvorba obsahu (napr. Blender, glTF modely).
- **E (Engine):** Softvérové prostredie pre správu scény a interakciu (napr. Unity, Three.js).
- **R (Runtime):** Systémové služby a API zabezpečujúce beh aplikácie (napr. **OpenXR**, WebXR).
- **H (Hardvér):** Cieľové koncové zariadenie (napr. Meta Quest).

4. Adaptivita rozhraní

Dôležitou koncepciou je **adaptivita**, čo je schopnosť rozhrania dynamicky v reálnom čase meniť svoju štruktúru a ovládanie podľa toho, aké zariadenie používateľ práve používa.

- **Štandardné adaptívne rozhrania** riadia komunikáciu medzi podsystémami (fyzika, senzory).
- **Viacpoužívateľské adaptívne rozhrania** umožňujú, aby v jednom prostredí spolupracovali ľudia s rôznym hardvérom (napr. jeden vo VR headsete a druhý cez webový prehliadač).

22) Proces interakcie v XR systémoch na jednotlivých stupňoch detailnosti.

Tento okruh sa zameriava na koncepčný a formálny rozklad toho, ako prebieha komunikácia medzi používateľom, výpočtovým jadrom a hardvérovými zariadeniami v systémoch rozšírenej reality. Proces interakcie je v zdrojoch rozdelený do troch úrovní detailnosti.

1. úroveň detailnosti: Základná slučka spätnej väzby

Na najvyššej úrovni je proces interakcie definovaný ako **uzavretý cyklus (feedback loop)**, v ktorom dáta prúdia medzi používateľom a systémom,. Tento model zahŕňa:

- **Používateľa:** Iniciuje aktivitu.
- **Vstup (Vstupný podsystém a rozhranie):** Zachytáva aktivitu používateľa a odovzdáva ju ďalej,.
- **Jadro systému:** Vykonáva logické operácie, transformácie a výpočty,.
- **Výstup (Výstupný podsystém a rozhranie):** Generuje odozvu, ktorú používateľ vníma,.
- **Zariadenie:** Hardvérový komponent, ktorý fyzicky sprostredkúva vstup a výstup,.

2. úroveň detailnosti: Sémantická a syntaktická rovina

Druhá úroveň detailnosti rozširuje proces o **formálne popisy údajov** a rozlišuje medzi tým, čo informácia znamená (sémantika), a ako je technicky zapísaná (syntax).

- **Diverzita:** Model zavádza deliacu čiaru „diverzity“ medzi používateľskými inštrukciami a technickým spracovaním v zariadení.
- **Vstupná časť:** Inštrukcia zariadenia (sémantická množina) sa transformuje do syntaktickej množiny $f(VS)$.

- **Spracovanie:** Prebieha v rámci **XR prostredia a kolaborácie**, kde sa stretáva syntaktická a sémantická množina údajov.
- **Výstupná časť:** Inštrukcia prostredia (sémantická) sa mení na syntaktickú množinu, definovanú funkciou $VT = f_M(f_S(f_Z(V S)))$, čo predstavuje finálnu transformáciu pre výstupné zariadenie.

3. úroveň detailnosti: Konkrétne technologická implementácia

Tretia úroveň detailnosti ukazuje, ako sa teoretické modely mapujú na reálny hardvér, pričom zdroje uvádzajú príklady pre VR systémy:

- **Príklad s bio-senzormi (EEG a Eye tracker):**
 - Snímače generujú **EEG a Eye tracker dáta** (syntaktická rovina).
 - Systém ich interpretuje ako **EEG a Eye tracker inštrukcie** (sémantická rovina).
 - Nasleduje **preklad sémantických inštrukcií** a ich aplikácia do virtuálneho prostredia, čo vedie k interakčnej odozve.
- **Príklad so snímaním gest (Leap Motion a RealSense):**
 - Senzory snímajú gestikuláciu alebo rekonštruujú reálnu scénu (syntax).
 - Systém tieto dáta chápe ako **sémantické gestá** alebo **kolízie**.
 - Výsledkom je inštrukcia prostredia vo forme **obrazu**, ktorý používateľ vidí na obrazovke.

Zhrnutie systému DEDSHO

Celý tento proces interakcie je zastrešený systémom **DEDSHO**, kde **Rozhranie (Interface)** funguje ako transformátor, ktorý buď zachováva sémantiku informácie, alebo ju prekladá medzi svetom používateľa a svetom stroja.

23) Konštrukcia voľne viazaných, tesne viazaných a kombinovaných XR systémov.

Konštrukcia XR systémov sa podľa miery previazania hardvérových prostriedkov s používateľom delí na tri základné kategórie: **tesne-viazané**, **voľne-viazané** a **kombinované** systémy.

1. Tesne-viazané systémy

Tieto systémy sú charakteristické tým, že zobrazovacie zariadenie je pevne uchytené priamo na tele používateľa, najčastejšie vo forme dátovej prilby (**HMD – Head-Mounted Display**).

- **Typy podľa zobrazovačov:**
 - **So zabudovanými zobrazovačmi:** Headset má vlastné displeje (napr. Meta Quest).
 - **Na báze smartfónov:** Zobrazovaciu a výpočtovú funkciu plní vložený mobilný telefón.
- **Konektivita:**
 - **Káblové prepojenie:** Tradičné riešenie, ktoré je lacnejšie, no obmedzuje pohyb používateľa.

- **Bezkáblové prepojenie:** Poskytuje vyššiu voľnosť pohybu (standalone zariadenia).
- **Vizualizácia:** Pre tieto systémy je typické **stereoskopické renderovanie** (samostatný obraz pre ľavé a pravé oko). Moderné trendy smerujú k **adaptívnym varifokálnym systémom**, ktoré riešia VAC konflikt (problém so zaostrovaním).

2. Voľne-viazané systémy

Sú to systémy, kde sa používateľ pohybuje vo vnútri stacionárnej konštrukcie, ktorá sprostredkúva virtuálny vnem. Najznámejším príkladom sú **virtuálne jaskyne (CAVE)**.

- **Technologické vyhotovenie:**
 - **Projekčné:** Obraz je premietaný na steny jaskyne pomocou dátových projektorov.
 - **Displejové:** Steny sú tvorené sústavou monitorov alebo veľkoplošných zobrazovačov.
- **Konštrukcia:** Pozostáva z **mechanickej nosnej konštrukcie**, ktorá obklopuje používateľa. Na rozdiel od HMD je tu **riadiaci výpočtový systém** spravidla realizovaný ako samostatná jednotka mimo samotnej konštrukcie.
- **Vizualizácia:** Využíva sa tzv. **viac-pohľadové (multi-view) renderovanie**, aby sa dosiahol správny vnem z rôznych uhlov pohľadu v rámci jaskyne.

3. Kombinované systémy (Hyper-realita)

Kombinované systémy predstavujú technologické spojenie oboch predchádzajúcich prístupov. Tento koncept sa označuje ako **Hyper-realita (HR)**.

- **Podstata:** Ide o **fyzickú kombináciu** prostredí a technológií. Príkladom je situácia, kedy sa používateľ vybavený tesne-viazaným zariadením (napr. MR okuliare HoloLens) nachádza vo vnútri voľne-viazaného systému (virtuálna jaskyňa VR).
- **Účel:** Cieľom je využiť výhody oboch riešení – napríklad vysokú imerziu jaskyne a schopnosť vkladať detailné interaktívne virtuálne objekty cez AR/MR headset.

Tieto konštrukčné prístupy sú v súčasnosti zastrešené jednotným priemyselným štandardom **OpenXR**, ktorý umožňuje vývojárom oddeliť softvérovú aplikáciu od konkrétneho hardvérového vyhotovenia.

24) Rozhrania XR systémov, rozhranie ako transformácia, bioadaptované a naturálne rozhrania.

Rozhranie (**interface**) predstavuje v systémoch rozšírenej reality kľúčový prvok, ktorý oddeľuje a zároveň spája digitálny svet s používateľom. Podľa technologickej úrovne rozlišujeme štyri základné kategórie rozhraní: **priame (mechanické)**, **znakové (CLI)**, **grafické (GUI)** a najmodernejšie **bioadaptované a prirodzené rozhrania**.

Rozhranie ako transformácia

V rámci systému interakcie vystupuje rozhranie ako **transformátor**, ktorý sprostredkováva tok dát medzi používateľom a jadrom systému. Tento proces transformácie možno popísať nasledovne:

- **Vstupná transformácia:** Inštrukcia používateľa (množina O) je transformovaná rozhraním na inštrukciu stroja (O').
- **Sémantika a syntax:** Rozhranie pri tomto procese narába s **prvkovou množinou a syntaxou** údajov.
- **Riadenie významu:** Transformácia môže prebiehať buď so **zachovaním sémantiky** (význam informácie ostáva rovnaký pre obe strany), alebo s **prekladom sémantiky** (rozhranie prekladá používateľský zámer do technickej reči stroja).

S procesom transformácie úzko súvisí aj **mediácia** (proces prevádzania údajov do formátu vhodného na vizualizáciu) a **medialita**, ktorá určuje mieru prepojenia medzi heterogénnymi svetmi používateľa a stroja.

Bio-adaptované rozhrania

Tieto rozhrania sú založené na priamej interakcii s biologickými procesmi živého organizmu. Ich podstata spočíva v dvoch rovinách:

1. **Snímanie (vstup):** Zachytávanie biologických funkcií používateľa. Príkladom sú **senzorické rozhrania** ako:
 - **EEG (Elektroencefalografia):** Snímanie aktivity mozgu, ktoré umožňuje ovládanie systému cieľovým myslením alebo mimickými vzormi.
 - **EMG (Elektromyografia):** Snímanie svalovej aktivity končatín (napr. náramok MYO).
 - **EOG (Elektrookulogram):** Snímanie svalovej aktivity očí pre ovládanie systému smerom pohľadu.
2. **Vytváranie vnemu (výstup):** Generovanie biologického vnemu priamo pre používateľa.

Tieto systémy vykazujú **vysokú mieru adaptivity**, kedy systém primerane reaguje na fyziologický stav používateľa.

Prirodzené (naturálne) rozhrania

Prirodzené rozhrania sú navrhnuté tak, aby bola interakcia so systémom čo **najkomfortnejšia a najbližšia reálnym ľudským vnemom**. Ich cieľom je **posilnenie pohltenia (imerzie)** používateľa do virtuálneho prostredia.

Charakteristické znaky a technológie:

- **Spôsob ovládania:** Využívajú reálne zaužívané spôsoby komunikácie, ako sú **gestá rúk** alebo **hlasové povely**.
- **Technické prostriedky:** Na snímanie sa využívajú zariadenia ako **MS Kinect, Leap Motion, OptiTrack** alebo rôzne typy dátových rukavíc.
- **Architektonická väzba:** Prirodzené rozhrania integrujú podporu viacerých podsystémov (vizualizačný, akustický, hmatový), aby vytvorili komplexný vizuálny, sluchový alebo hmatový vnem.
- **Interakčné techniky:** Umožňujú prirodzené vykonávanie aktivít, ako je **selekcia** (označovanie), **manipulácia** (presúvanie, otáčanie) a **lokomícia** (pohyb v priestore).

Všetky typy rozhraní môžu byť ďalej implementované ako **adaptívne**, čo im umožňuje v reálnom čase meniť svoju štruktúru a funkcionálnosť podľa typu použitého hardvéru alebo potrieb kolaborácie.

25) Adaptívne rozhrania XR systémov a proces adaptácie vstupov a výstupov.

1. Definícia a účel adaptivity

Adaptivita je vlastnosť rozhrania, ktorá mu umožňuje **prispôbiť svoje ovládanie a štruktúru** v závislosti od typu konkrétného používateľského zariadenia. Jej hlavným účelom v XR systémoch je zabezpečiť také **interakčné techniky**, ktoré zodpovedajú možnostiam hardvéru a rozhraní, ktoré má používateľ momentálne k dispozícii.

2. Teoretické definície adaptívnych rozhraní

Zdroje uvádzajú tri dôležité pohľady na adaptívne rozhrania:

- **Gulla:** Definuje ich ako systémy schopné meniť svoju štruktúru a funkcionálnosť podľa okolností používania v **reálnom čase**.
- **Roldán:** Považuje ich za podsystemy, ktoré **dynamicky detegujú vstupy** používateľov a následne ich priradujú ku konkrétnej funkcionálite ovládania.
- **Castillejo:** Reprezentuje ich ako **inteligentné komunikačné rozhrania**, ktorých cieľom je uľahčiť interakciu.

3. Proces adaptácie vstupov na výstupy

Samotný proces, ktorým adaptívne rozhranie transformuje používateľský príkaz na systémovú aktivitu a spätnú väzbu, prebieha v nasledovných logických krokoch:

1. **Vstupný príkaz:** Používateľ iniciuje aktivitu (napr. gesto alebo stlačenie tlačidla).
2. **Rozpoznanie systému/podsystemu:** Rozhranie identifikuje kontext a zariadenie, z ktorého príkaz prišiel.
3. **Rozpoznanie a adaptácia vstupu:** Systém analyzuje vstup a pomocou knižnice **komponentov (funkcií)** mu priradí zodpovedajúcu funkcionálnosť.
4. **Aplikovanie funkcionality:** Vykoná sa preklad vstupu na konkrétnu akciu v jadre XR systému.
5. **Vytvorenie aktivity a určenie spätnej väzby:** Na základe vykonanej akcie systém určí formu odpovede.
6. **Výstupná odpoveď:** Používateľ dostane informáciu o výsledku aktivity cez výstupné podsystemy (obraz, zvuk, haptika).

4. Komponenty a dynamika rozhraní

Adaptívne rozhrania sú modulárne a pozostávajú z rôzneho počtu **komponentov**, ktoré sa kombinujú podľa potreby adaptácie konkrétného vstupu. Významnou vlastnosťou je, že **počet a zloženie týchto komponentov sa môže meniť dynamicky v čase**, čo umožňuje systému reagovať napríklad na zmenu hardvéru počas behu aplikácie.

5. Adaptácia interakčných techník podľa typu aktivity

Proces adaptácie rozlišuje vstupné aktivity používateľa podľa ich časového trvania:

- **Statické aktivity:** Sú detegované v krátkom okamihu ako jednorazová udalosť (stlačenie tlačidla, statické gesto). Adaptujú sa primárne na techniku **Selekcie** (označenie objektu, teleportácia).
- **Dynamické aktivity:** Prebiehajú v čase a vyžadujú kontinuálne snímanie. Adaptujú sa na techniky **Manipulácie** (presúvanie, rotácia objektov) a **Lokomócie** (plynulý pohyb v prostredí).

Tento proces zabezpečuje, že či už používateľ používa dotykové rozhranie, haptické ovládače alebo snímanie gest, výsledná interakcia v zdieľanom virtuálnom prostredí je konzistentná.

26) Adaptívne rozhrania XR systémov podľa účelu nasadenia.

Adaptívne rozhrania v systémoch rozšírenej reality sa podľa účelu ich nasadenia klasifikujú do dvoch základných kategórií: **štandardné adaptívne rozhrania XR podsystémov** a **viacpoužívateľské adaptívne rozhrania XR systémov**.

1. Štandardné adaptívne rozhrania XR podsystémov

Tieto rozhrania slúžia ako prostredník medzi kolaboratívnym virtuálnym prostredím a konkrétnymi hardvérovými či softvérovými podsystémami. Ich hlavné úlohy a charakteristiky sú:

- **Riadenie komunikácie:** Zabezpečujú tok dát medzi jadrom systému a **simulačným, fyzikálnym a senzorickým podsystémom**.
- **Funkcie:** Zodpovedajú za podávanie vstupov a výstupov, príkazy ovládania a riadenie vlastností samotného virtuálneho prostredia.
- **Roviny spracovania:** Implementácia prebieha na troch úrovniach:
 - **Úroveň komunikácie** rozhrania s konkrétnymi podsystémami.
 - **Úroveň spracovania údajov** (data processing).
 - **Úroveň replikácie** spracovaných údajov smerom k používateľom.

2. Viacpoužívateľské adaptívne rozhrania XR systémov

Tento typ rozhraní je navrhnutý pre potreby **multiplatformovej kolaborácie**, kde používatelia pristupujú do spoločného priestoru z rôznych typov zariadení (napr. jeden cez VR headset, iný cez mobilnú AR).

- **Integrácia platforiem:** Zabezpečujú zjednotenú interakciu naprieč technológiami VR, MR a AR.
- **Rozpoznávanie funkcionality:** Kľúčovou vlastnosťou je **úroveň rozpoznávania dostupnej funkcionality zariadení**, kedy systém zisťuje, aké interakčné možnosti má konkrétny používateľ k dispozícii a podľa toho mu prispôsobí ovládanie.
- **Riadenie interakcie:** Rozhranie adaptívne priradzuje vstupné príkazy používateľa (z rôznych ovládačov či senzorov) k jednotným interakčným technikám, ako sú **selekcia, manipulácia a lokomócia**.

Využitie týchto adaptívnych rozhraní je prínosné najmä vo webových platformách, smart prostrediach a systémoch využívajúcich umelú inteligenciu, pretože umožňujú systému dynamicky v reálnom čase meniť svoju štruktúru podľa aktuálnych potrieb a hardvéru.

27) XR a bio-senzorické rozhrania.

1. Podstata a účel bio-senzorických rozhraní

Bio-senzorické rozhrania v XR systémoch slúžia na **priame získavanie fyziologických vstupov**, ktoré používateľ produkuje. Ich hlavné využitie spočíva v systémoch určených na **kontrolu a monitorovanie fyziologického stavu** používateľa. Tieto systémy sa vyznačujú **vysokou adaptivitou**, kedy systém dokáže v reálnom čase primerane reagovať na zmeny v nameraných vstupoch z bio-senzorov.

2. Sledované fyziologické parametre

Súčasná technológia umožňuje v rámci XR systémov snímať a vyhodnocovať rôzne biologické prejavy:

- **Neurónová a svalová aktivita:** Elektroencefalografia (**EEG**) a elektromyografia (**EMG**).
- **Srdcová činnosť a krv:** Elektrokardiografia (**EKG**), krvný tlak a hladina kyslíka v krvi.
- **Stav pokožky:** Teplota, hydratácia, pružnosť kože a citlivosť tkanív.
- **Metabolické údaje:** Hladina glukózy a minerálov v tele.

3. Detailný pohľad na EEG (Elektroencefalografia)

EEG rozhrania snímajú aktivitu mozgu používateľa. V rámci virtuálnej reality sa využívajú na:

- **Ovládanie mimickými vzormi:** Systém reaguje na pohyby svalov tváre zachytené cez senzory na hlave.
- **Cielené myslenie:** Používateľ môže ovplyvňovať prostredie sústredením sa na vopred definované pravidlá, napríklad na tvary alebo farby.

4. Proces spracovania v bio-adaptívnom rozhraní

Interakcia cez bio-senzory prebieha v rámci systému detailnosti údajov nasledovne:

1. **Syntaktická rovina:** Senzory generujú surové dáta (napr. EEG dáta alebo dáta o svalovom napätí).
2. **Sémantická rovina:** Systém tieto dáta interpretuje ako konkrétne **biometrické inštrukcie**.
3. **Preklad a aplikácia:** Inštrukcie sa preložia do logiky virtuálneho prostredia, čo vedie k **interakčnej odozve**, ktorú používateľ vníma napríklad vo forme zmeny obrazu.

5. Technologické a konštrukčné zaradenie

Bio-senzorické rozhrania sú z hľadiska technológií často implementované ako **bio-adaptované rozhrania**. Ich cieľom je vytvoriť čo najužšie prepojenie medzi živým organizmom a strojom, pričom môžu fungovať nielen ako vstup (snímanie), ale teoreticky aj ako výstup (vytváranie biologického vnemu). V praxi sa často kombinujú s prvkami **prirodzených rozhraní**, ako je sledovanie očí.

28) Programová implementácia XR systému a jej hierarchia, real-time obmedzenia, procesný model implementácie.

1. Hierarchia programovej implementácie XR systému

Programová implementácia XR systému je v zdrojoch rozdelená do troch základných úrovní detailnosti, ktoré postupujú od abstraktného návrhu ku konkrétnej vizualizácii:

1. **Popis logického jadra:** Ide o najvyššiu úroveň, kde sa formálne definuje logika systému pomocou grafov, diagramov alebo iných formalizmov (napr. Petriho siete). Stanovujú sa tu **záujmové oblasti**, ich parametre a kľúčové vzťahy medzi nimi.
2. **Modelovanie XR (popisné prostriedky):** Na tejto úrovni sa využíva formálny návrh na popis virtuálneho priestoru. Využívajú sa:
 - **Deskriptívne prostriedky:** Pre popis scény a objektov (napr. **USD**, **glTF**, historicky **VRML**).
 - **Skriptovacie prostriedky:** Pre definovanie transformácií a správania v reálnom čase (napr. **Python**, **LUA**, **C#**, **C++** alebo vizuálne skriptovanie ako **Blueprinty**).
3. **Vizualizácia / Simulácia:** Najnižšia úroveň, ktorá predstavuje finálne rozhranie, v ktorom používateľ sleduje a interaktívne ovplyvňuje prebiehajúce procesy (virtuálne objekty, terény, senzory).

2. Real-time obmedzenia XR systémov

Keďže XR systém musí pracovať v reálnom čase, je kriticky závislý od dodržiavania časových limitov a deterministického správania. Implementácia prebieha v troch základných slučkách: **simulačnej**, **renderovacej** a **interakčnej**.

Kľúčové časové obmedzenia zahŕňajú:

- **Obnovovacia frekvencia:** Musí sa udržiavať stabilne v rozmedzí **60 – 120 Hz**.
- **Časový rozpočet snímky (frame budget):** Maximálny čas, ktorý má systém na výpočet a vykreslenie jedného snímku, aby nedošlo k sekaniu obrazu.
- **Dôsledky nedodržania:** Ak systém nestíha, dochádza k zníženiu imerzie, nestabilite a u používateľa nastupuje **cybersickness (kybernetická nevoľnosť)**.

3. Procesný model implementácie

Implementácia XR systému je viac-vrstvový proces postavený na tzv. **virtualizačnom reťazci**, ktorý zahŕňa tieto kroky:

1. Zber a analýza požiadaviek.
2. Formalizácia systému (popis logického jadra).
3. Tvorba a modelovanie virtuálneho priestoru a objektov (assets).
4. Simulácia procesov.
5. Vizualizácia, interakcia a spätná väzba používateľa.

Tento procesný model sa v praxi realizuje cez **XR rúru (pipeline)**, čo je životný cyklus spracovania dát v reálnom čase. Táto pipeline integruje logické jadro, assety, fyzikálne moduly a renderovací subsystém.

4. Implementačné stupne (Programové prostriedky)

Z pohľadu IT prostriedkov zahŕňa programová implementácia najmä:

- **Definičné prostriedky:** Pre statické a dynamické objekty a scény.
- **Operačný systém a jadro:** Riešenie matematických transformácií a detekcie kolízií.
- **Dátové štruktúry:** Spôsoby opisu modelov a assetov.
- **Skriptovanie:** Pre oživenie logiky a interakcie používateľa.

Základom úspešnej implementácie je rovnováha medzi technologickými možnosťami (presnosť, výkon) a rešpektovaním ľudských biologických limitov.

29) Postup spracovania požiadaviek na XR systém.

Postup spracovania požiadaviek na XR systém predstavuje úvodnú fázu procesu implementácie a je kľúčový pre správnu **formalizáciu virtuálneho priestoru**. Tento postup pozostáva z piatich logických krokov:

1. Definovanie oblastí záujmu

Prvým krokom je určenie oblastí záujmu, ktoré vychádzajú zo zamerania systému – teda pre koho je určený a akú má mať funkčnosť. Tieto oblasti sa delia na dve skupiny:

- **Statické objekty:** Definujú samotný priestor vo virtuálnej realite (napr. budovy, cesty).
- **Dynamické objekty:** Definujú procesy prebiehajúce v systéme (napr. pohybujúci sa ľudia alebo dopravné prostriedky).

2. Určenie parametrov pre oblasti záujmu

Pre každú definovanú oblasť záujmu je potrebné stanoviť konkrétne parametre, ktoré môžu byť:

- **Interné (lokálne):** Parametre potrebné priamo v rámci výpočtového systému.
- **Externé (globálne):** Parametre ovplyvňujúce širší kontext prostredia.

3. Otypovanie parametrov

Parametre sa nerozdeľujú len podľa programátorských dátových typov, ale podľa ich funkčného významu v systéme do štyroch kategórií:

- **Identifikátory:** Parametre, ktoré objekt identifikujú (napr. názov), ale ich zmena neovplyvňuje iné parametre.
- **Logické hodnoty:** Parametre ovplyvňujúce iné časti systému (napr. podmienka „potreba lekárskeho predpisu“).
- **Číselné hodnoty:** Hodnoty, ktoré ovplyvňujú systém alebo sú ním ovplyvňované (napr. zisk z predaja).

- **Množiny hodnôt:** Skupiny hodnôt s vplyvom na systém (napr. zoznam účinných látok v lieku).

4. Definovanie vzťahov medzi parametrami

V tomto kroku sa určuje, ako jednotlivé parametre medzi sebou interagujú. Vzťahy môžu byť **medzi údajové, medzi inštrukčné alebo hybridné**. Zároveň sa delia na **procedurálne a neprocedurálne**. Pri analýze vzťahov sa môže ukázať, že pôvodne dôležité parametre sú v skutočnosti len identifikátormi bez vplyvu na iné časti systému.

5. Určenie typov pre jednotlivé vzťahy

Každému vzťahu je priradený typ závislosti, ktorý určuje dynamiku systému:

- **Priamoúmerná závislosť:** Napríklad zvýšením ceny lieku sa zvýši zisk.
- **Nepriamoúmerná závislosť:** Napríklad zvýšením nákupnej ceny klesá zisk.
- **Podmienená závislosť:** Predstavuje existenčnú závislosť dvoch parametrov – ak nie je splnená podmienka (napr. chýbajúci predpis), proces v systéme nemôže existovať.

Význam tohto postupu: Správne spracovanie požiadaviek umožňuje používateľovi **sledovať zmeny v systéme** pri úprave vstupných dát a zároveň **identifikovať vplyvy**, ktoré vedú k sledovanému výstupu systému.

30) Virtuálny priestor a jeho parametre.

1. Definícia virtuálneho priestoru

Virtuálny priestor (alebo virtuálny svet) je v zdrojoch definovaný ako **základná entita**, v ktorej sa odohráva a realizuje virtuálny dej alebo život. Predstavuje prostredie, v ktorom sú umiestnené virtuálne objekty a v ktorom prebieha interakcia s používateľom.

2. Základné charakteristiky priestoru

Každý virtuálny priestor je charakterizovaný štyrmi hlavnými aspektmi:

- **Dimenzia:** Rozmer priestoru (napr. 2D, 3D), pričom zdroje varujú pred nežiaducim miešaním štruktúr rôznych dimenzií.
- **Súradnicová sústava:** Matematický rámec pre určovanie polohy objektov.
- **Definičná úroveň:** Formálny opis priestoru a objektov, ktoré sa v ňom nachádzajú.
- **Dynamika:** Zahŕňa virtuálne plynutie času, pohyby, transformácie a deje.

3. Klasifikácia priestoru podľa parametrov

Zdroje rozdeľujú virtuálne priestory podľa viacerých technických a matematických kritérií:

A. Podľa typu dimenzie

1. **Celočíselné (topologické):** Štandardné priestory s celým počtom rozmerov (0, 1, 2, 3...).
2. **Neceločíselné:** Napríklad Hausdorffova alebo fraktálna dimenzia (napr. rozmer 2,6), ktoré popisujú zložité štruktúry.

B. Podľa charakteru priestoru

1. **Translačný priestor:** Zameraný na posun v priestore.
2. **Rotačný priestor:** Zameraný na otáčanie objektov.
3. **Kombinovaný priestor:** Spája oba predchádzajúce prístupy.

C. Podľa štruktúry dimenzií

1. **Homogénna štruktúra ($N+0$):** Bežný N-rozmerný priestor (napr. 3D priestor).
2. **Heterogénna štruktúra ($N_1+N_2+...+N_n$):** Zložený priestor kombinujúci rôzne typy rozmerov.

D. Podľa počtu stupňov voľnosti (DOF – Degree of Freedom)

DOF vyjadruje počet nezávislých parametrov, ktoré definujú stav objektu alebo avatara v priestore. Určuje, v koľkých smeroch sa môže entita pohybovať (translačne, rotačne alebo časovo). Delí sa na:

1. **Jednosmerné:** Pohyb je možný len v jednom smere danej dimenzie.
2. **Obojsmerné:** Pohyb je možný v oboch smeroch dimenzie.

4. Príklady špecifikácie priestorov

Kombináciou týchto parametrov vznikajú konkrétne typy prostredí, ako napríklad:

- **Celočíselný heterogénny obojsmerný translačný priestor.**
- **Celočíselný homogénny jednosmerný rotačný priestor.**

5. Rozmer (dimenzia) podrobnejšie

Rozmer môže byť číselný alebo nečíselný. Okrem topologickej metriky zdroje uvádzajú aj dimenziu **sebepodobnosti** (fraktálnu), **informačnú dimenziu** alebo **dimenziu rotácií**. Štruktúra a adekvátne dimenzionalita objektov je vždy priamo závislá od štruktúry a homogenity virtuálneho sveta, v ktorom sú umiestnené.

31) Reprezentácia a popis virtuálneho sveta a objektov, elementy, vlastnosti, vzťahy, modely, formalizačné prostriedky opisu virtuálneho sveta, mapovanie formálneho modelu.

1. Reprezentácia virtuálneho sveta a jeho elementy

Virtuálny svet je v systémoch XR reprezentovaný ako súbor zoznamov rôznych entít, ktoré spolu tvoria koherentné prostredie. Medzi základné **elementy** patrí:

- **Zoznam máp, terénov a povrchov.**
- **Zoznam senzorov, kamier a kolidrov** (objektov slúžiacich na detekciu zrážok).
- **Zoznam virtuálnych objektov**, ktoré môžu byť statické alebo dynamické.
- **Zoznam objektov iného typu údajov.**

Tento koncept reprezentácie sa využíva ako fundamentálny základ pre **entitno-komponentné systémy (ECS)**, simulačné systémy a pri definovaní digitálnych dvojčiat.

2. Vlastnosti objektov

Každý objekt vo virtuálnom svete má definovanú svoju štruktúru prostredníctvom parametrov. Tieto vlastnosti sa delia na:

- **Globálne parametre:** Spoločné pre celý objekt, ako je poloha, materiál, rozmery, tvar a časové charakteristiky.
- **Lokálne parametre (rámce):** Špecifické vlastnosti rozdelené podľa vnemových kategórií:
 - **Vizualizačný rámec:** Farba a optické vlastnosti.
 - **Akustický rámec:** Zvuk objektu, zvuk pri kolízii a odraz zvuku.
 - **Kinematický a statokinetický rámec:** Deformácia objektu a jeho pohybové limity.
 - **Hmatový rámec:** Sila vibrácií a sila spätnej väzby.
 - **Ostatné rámce:** Vôňa, chuť či citlivosť na feromóny.

3. Vzťahy v reprezentácii sveta

Vzťahy určujú, ako medzi sebou jednotlivé elementy (napr. senzor a virtuálny objekt) komunikujú prostredníctvom **údajov a inštrukcií**.

- Vzťahy **nemusia byť stále**; môžu vznikať a zanikať v závislosti od situácie.
- Môžu byť **časovo podmienené** alebo **reaktívne** (aktivujú sa ako reakcia na udalosť – event-driven).

4. Modely a ich kombinácie

Pre zložitejšie entity, ako sú organizmy, sa využíva kombinácia viacerých modelov, ktoré tvoria jeden celok. Typicky ide o **skeletálny (kosti), muskulárny (svalstvo), povrchový (koža), fyzikálny, riadiaci (AI), emocionálny a kinematický model**. Pri vytváraní odvodených typov objektov v systéme ECS je nutné, aby boli sady parametrov týchto modelov kompatibilné alebo prepojené relačnými funkciami.

5. Formalizačné prostriedky opisu

Slúžia na presný technický popis virtuálneho priestoru a jeho dynamiky.

- **Deskriptívne (popisné) prostriedky:** Slúžia na opis štruktúry scény a vzťahov medzi entitami.
 - *Historické:* **VRML**, OpenInventor, XML-based popisy.
 - *Súčasná:* **USD** (Universal Scene Description) a **glTF** (optimalizovaný pre runtime a distribúciu 3D dát).
- **Skriptovacie (procedurálne) prostriedky:** Slúžia na popis správania, transformácií a simulácií. Príkladom sú **Python, LUA, C#, C++** alebo vizuálne skriptovanie vo forme **Blueprintov**.

6. Mapovanie formálneho modelu na XR systém

Formálny návrh sa v praxi mapuje na konkrétne softvérové súčasti systému:

- **Logické/procesné jadro (XR core)** rieši výpočtovú logiku.

- **Virtuálny priestor/scéna (scene graph)** definuje hierarchické usporiadanie sveta.
- **Transformačný reťazec** zabezpečuje geometrické zmeny a rotácie, ktoré sú dnes primárne implementované na báze **quaterniónov**.

Implicitná implementácia týchto formálnych modelov v XR systémoch umožňuje **systematický návrh, analýzu chýb a lepšiu kontrolu správania celého systému**.

32) Súradnicové sústavy, súradnicové sústavy vyšších rádov, vzťahy priestorov, transformácie.

1. Súradnicové sústavy vo virtuálnom priestore

Súradnicová sústava je základným matematickým rámcom, ktorý charakterizuje každý virtuálny priestor. Umožňuje jednoznačne určiť polohu bodov a objektov. Podľa charakteru pohybu a priestoru ich delíme na:

- **Translačná sústava:** Zameraná na posun v smeroch osí (napr. x , y , z).
- **Rotačná sústava:** Zameraná na určovanie orientácie a otáčania pomocou uhlov.
- **Kombinovaná sústava:** Spája oba predchádzajúce prístupy pre plný popis stavu objektu.

Dôležitým parametrom sú **stupne voľnosti (DOF – Degree of Freedom)**, ktoré vyjadrujú počet nezávislých parametrov (translačných, rotačných alebo časových) definujúcich stav objektu v danej sústave.

2. Súradnicové sústavy vyšších rádov

V kontexte XR systémov sa pod sústavami vyšších rádov rozumejú najmä systémy na báze **quaterniónov**.

- **Quaternióny:** Predstavujú 4D súradnicový systém použitý na reprezentáciu rotácií v 3D priestore.
- **Výhody:** Sú **nezávislé od všeobecného súradnicového systému**, majú zreteľnú geometrickú interpretáciu a zamedzujú strate stupňa voľnosti, známej ako **gimbal lock**.

3. Vzťah priestorov a sústav

Z hľadiska návrhu je dôležité, že **transformácia sa v rôznych sústavách môže prejavovať rozdielne**.

- **Zjednodušenie popisu:** Rôzne manipulácie s tým istým objektom je často jednoduchšie popísať v inej sústave (napr. lokálnej namiesto svetovej) a následne vyjadriť transformáciu medzi nimi.
- **Homogenita:** Štruktúra objektu a jeho dimenzionalita musí byť v súlade s homogenitou virtuálneho sveta (geometria, čas a pod.), aby nedochádzalo k nežiaducemu miešaniu štruktúr.

4. Transformačný reťazec

Proces premeny objektu z jeho základného popisu až po zobrazenie na displeji prebieha v rámci tzv. **transformačného reťazca**, ktorý prepája rôzne sústavy pomocou transformačných matic:

1. **SSO (Súradnicová sústava objektu):** Lokálny priestor, v ktorom je objekt definovaný.

2. **GT (Geometrická transformácia):** Umiestnenie objektu do scény.
3. **USS (Univerzálna súradnicová sústava):** Globálny "svetový" priestor scény.
4. **NSS (Normalizovaná súradnicová sústava):** Medzistupeň pre spracovanie viditeľnosti a orezávania.
5. **PT (Projekčná transformácia):** Prepočet 3D scény do 2D pohľadu kamery.
6. **SSC (Súradnicová sústava kamery):** Pohľad z perspektívy pozorovateľa (oka).
7. **ZT (Zobrazovacia transformácia):** Príprava dát pre konkrétny zobrazovač.
8. **SSZ (Súradnicová sústava zobrazovača):** Finálna logická a fyzická sústava pixelov na displeji (napr. v headsete).

5. Typy transformácií

Transformácie (T), ktoré menia objekt O na O' , delíme na:

- **Lineárne:** Zachovávajú priamky a rovnobežnosť (posun, rotácia, škálovanie).
- **Nelineárne:** Spôsobujú deformácie (napr. perspektívne skreslenie alebo ohýbanie).

V súčasnosti sa pre výpočet rotácií v tomto reťazci využívajú primárne matematické knižnice na báze **quaterniónov**, ktoré zabezpečujú plynulosť a stabilitu výpočtov v reálnom čase.

33) Quaternióny – súradnicový systém, Eulerove uhly, Eulerov teorém, základné pojmy.

1. Metódy definovania rotácie

V systémoch virtuálnej reality sa na popis otáčania objektov v priestore využívajú dve základné metódy:

- **Otáčanie definované Eulerovými uhlami:** Je reprezentované všeobecnými transformačnými maticami.
- **Otáčanie definované Eulerovým teorémom:** Je reprezentované pomocou **quaterniónov**.

Je dôležité rozlišovať medzi **orientáciou** (daná polohovým vektorom voči stredu sústavy) a **rotáciou** (otočením), ktorá je definovaná osou a uhlom otočenia.

2. Eulerov teorém

Eulerov teorém tvorí teoretický základ pre prácu s rotáciami v 3D priestore. Hovorí, že ak máme dve ľubovoľné orientácie O a O' , potom **existuje os l a uhol rotácie α** taký, že orientácia O prejde na O' , ak sa otočí o uhol α okolo tejto osi l .

- Tento výsledok konštatuje existenciu takejto osi a uhla, no nie ich jedinečnosť.
- Pre úplný popis rotácie (nasledovanej prípadnou zmenou mierky) sú potrebné štyri čísla: jedno pre mierku, jedno pre uhol a dve pre určenie roviny rotácie.

3. Quaternióny – základné pojmy

Quaternióny (objavené W. R. Hamiltonom v roku 1843) sú matematickým rozšírením komplexných čísel. Zatiaľ čo komplexné číslo má jednu reálnu a jednu imaginárnu zložku, quaternión pozostáva zo štyroch zložiek:

- **s (alebo w):** Reálna zložka.
- **i, j, k:** Tri imaginárne zložky.

Pre imaginárne zložky platia fundamentálne vzťahy:

- $i^2 = j^2 = k^2 = ijk = -1$
- $i \times j = k; j \times k = i; k \times i = j$ (pričom násobenie nie je komutatívne: $i \times j = -j \times i$).

4. Formáty zápisu quaterniónov

V praxi sa využívajú tri hlavné formáty zápisu:

1. **Lineárna kombinácia:** $q = w + xi + yj + zk$.
2. **Vektor štyroch koeficientov:** $[x, y, z, w]$.
3. **Dvojica (skalár, vektor):** (s, v) , kde s je reálna časť (w) a v je vektor imaginárnych častí $[x, y, z]$.

5. Základné operácie s quaterniónmi

- **Opačný quaternión:** $q' = -q = -w - xi - yj - zk$.
- **Veľkosť (absolútna hodnota):** $|q| = \sqrt{w^2 + x^2 + y^2 + z^2}$.
- **Inverzný quaternión:** Definovaný ako $q^{-1} = q' / (q \times q')$.
- **Násobenie:** Využíva sa na výpočet rotácie, pričom quaternióny sú **asociatívne**, ale **nie sú komutatívne** ($q_1 \times q_2 \neq q_2 \times q_1$).

6. Výhody oproti maticovej reprezentácii

Hoci jedinou výhodou matíc je schopnosť reprezentovať všetky typy transformácií naraz, quaternióny majú v XR systémoch v rámci rotácií tieto prednosti:

- **Zreteľná geometrická interpretácia** a nezávislosť od súradnicového systému.
- **Kompaktná reprezentácia** (vyžadujú menej pamäte ako matice).
- **Zamedzenie straty stupňa voľnosti (gimbal lock)**, čo je kritické pre stabilitu zobrazenia.
- **Jednoduchá kompozícia** viacerých rotácií za sebou.

34) Quaternióny a rotácia, rotácia okolo všeobecnej priamky, gimbal lock, definícia.

1. Quaternióny a výpočet rotácie

V systémoch virtuálnej reality sa bod v priestore $B[x_B, y_B, z_B]$ reprezentuje pomocou quaterniónu s nulovou reálnou zložkou: $\mathbf{B} = (0, \mathbf{B})$. Samotný výpočet rotácie tohto bodu pomocou rotačného quaterniónu q prebieha podľa rovnice:

$$\mathbf{B}_{\text{rotovany}} = \mathbf{q} \mathbf{B} \mathbf{q}^{-1}.$$

Kde:

- $\mathbf{q}$ je rotačný quaternion definujúci smer a uhol.
- $\mathbf{q}^{-1}$ je inverzný quaternion.
- Výsledkom je nový vektor polohy po otočení.

Hoci quaterniony pracujú v 4D priestore, pre potreby vykresľovacieho reťazca (GPU) je možné z quaternionu $\mathbf{q}$ kedykoľvek vygenerovať **transformačnú maticu rotácie $\mathbf{Q}$** (rozmerov 4x4), ktorá sa následne aplikuje ako štandardná rotačná matica.

2. Rotácia okolo všeobecnej priamky

Zdroje rozlišujú dva prístupy k rotácii okolo ľubovoľnej osi (priamky) v priestore:

- **Na báze Eulerových uhlov:** Tento proces je matematicky zložitý a vyžaduje si reťazec siedmich transformácií: posun priamky do počiatku (T_P), sériu rotácií pre stotožnenie priamky s jednou zo súradnicových osí (T_{Oa} , T_{Ob}), samotnú rotáciu (T_O) a následne vykonanie inverzných operácií v opačnom poradí (T_{Ob}^{-1} , T_{Oa}^{-1} , T_P^{-1}).
- **Na báze Eulerovho teorému (Quaterniony):** Tento prístup je priamočiarejší. Eulerov teorém hovorí, že každá zmena orientácie sa dá popísať ako **jeden uhol α okolo jednej osi l** . Quaterniony túto os a uhol integrujú priamo do svojej štruktúry, čím odpadá potreba zložitého skladania transformačných matíc.

3. Gimbal lock (Strata stupňa voľnosti)

Gimbal lock je kritický technický problém, ktorý vzniká pri používaní **Eulerových uhlov** (rotácie podľa osí X, Y, Z).

- **Podstata:** Nastáva v situácii, kedy sa dve z troch rotačných osí dostanú do paralelnej polohy, čím systém **stráca jeden stupeň voľnosti (DOF)**.
- **Dôsledok:** Používateľ nemôže otáčať objektom v určitom smere, kým nezmení orientáciu v iných osiach, čo v XR systémoch spôsobuje trhanie obrazu a nestabilitu.
- **Riešenie:** Quaterniony sú **imúnne voči Gimbal locku**, pretože reprezentujú rotáciu v priestore plynule bez závislosti od poradia osí.

4. Definícia a atribúty

- **Definícia rotácie:** Na rozdiel od orientácie (vektor polohy), rotácia (otočenie) je v priestore R^3 definovaná **osou a uhlom otočenia**.
- **Popis rotácie:** Podľa Hamiltona (1843) sú pre úplný popis rotácie (prípadne so zmenou mierky) potrebné **štyri čísla**:
 1. Jedno číslo pre zmenu mierky.
 2. Jedno číslo pre veľkosť uhla α .
 3. Dve čísla pre určenie roviny, v ktorej rotácia prebieha.

Práve táto štvorica parametrov tvorí základnú dátovú štruktúru quaterniónu, ktorá je **kompaktnejšia a výpočtovo stabilnejšia** než maticová reprezentácia.

35) Virtualizačný reťazec a jeho realizácia, práca s virtuálnym svetom, vizualizačný subsystém, sledovanie polohy, zobrazovacie systémy, 3D tlač a vytváranie reálnych objektov.

1. Virtualizačný reťazec a jeho realizácia

Virtualizačný reťazec predstavuje viacvrstvový procesný model implementácie XR systému.

Pozostáva z piatich nadväzujúcich krokov, ktoré tvoria uzavretý cyklus medzi reálnym a virtuálnym svetom:

1. **Príprava a získavanie vstupných údajov:** Zber požiadaviek a analýza dát z reálneho sveta.
2. **Modelovanie, editácia a formalizácia:** Tvorba assetov, definovanie logického jadra a statických/dynamických objektov.
3. **Overovanie:** Kontrola správnosti modelov a vzťahov pomocou spätnej väzby.
4. **Spracovanie, vizualizácia a práca s virtuálnym svetom:** Hlavná fáza interakcie používateľa s prostredím.
5. **3D tlač a vytváranie reálnych objektov:** Proces návratu z digitálnej formy do fyzickej reality.

Tento reťazec sa v praxi realizuje cez tzv. **XR rúru (pipeline)**, čo je životný cyklus spracovania dát v reálnom čase.

2. Práca s virtuálnym svetom

Práca s virtuálnym svetom predstavuje štvrtý stupeň reťazca, kde dochádza k **simulácii procesov a interakcii**. V tejto fáze používateľ cez rozhranie sleduje a interaktívne ovplyvňuje dej. Z technického hľadiska tu prebieha spracovanie virtuálnych objektov a polygónov v rámci vizualizačného jadra.

3. Vizualizačný podsystém (PG)

Je kľúčovým podsystémom, nakoľko človek spracúva približne **80 % informácií zrakom**. Jeho úlohou je transformovať popis modelu sveta do výstupného obrazu.

Vrstvy vizualizačného procesu zahŕňajú:

- Definovanie a spracovanie modelu (reprezentácia, súradnicové sústavy).
- Geometrické transformácie nad objektami.
- Riešenie viditeľnosti v 3D a 2D.
- Tieňovanie, osvetľovanie a zrkadlenie.
- Rasterizácia a kompozícia finálneho obrazu do zásobníka snímkov.

Špecifické požiadavky pre XR: Kvôli stabilite vnemu musí systém udržiavať vysokú obnovovaciu frekvenciu (**90–120 Hz**) a minimálnu latenciu (**motion-to-photon**).

4. Sledovanie polohy (Tracking)

Sledovanie polohy je nevyhnutné pre synchronizáciu pohybov používateľa s virtuálnou scénou.

- **Technológie:** Využívajú sa mechanické, magnetické, ultrazvukové, laserové alebo optické snímače.
- **Metódy:** Moderné systémy (najmä markerless) využívajú metódu **SLAM** (Simultaneous Localization and Mapping), ktorá simultánne určuje polohu zariadenia a mapuje prostredie pomocou kamier a IMU senzorov.
- **Zariadenia:** Historické i súčasné príklady zahŕňajú **MS Kinect, Leap Motion, Optitrack** či systémy pre motion capture celého tela ako **Teslasuite**.

5. Zobrazovacie systémy

Zariadenia sa delia podľa konštrukcie na:

- **Tesne-viazané (HMD):** XR headsety ako **Meta Quest, MS HoloLens, Apple Vision Pro** či **HTC Vive**. Smerujú k adaptívnym varifokálnym displejom, ktoré riešia VAC konflikt.
- **Voľne-viazané:** Virtuálne jaskyne (**CAVE**), kde je obraz premietaný na steny stacionárnej konštrukcie.
- **Technológie zobrazenia:** Stereoskopické systémy (aktívne/pasívne), priestorové displeje (Volumetric), holografické systémy alebo projekcia na sieťnicu.

6. 3D tlač a vytváranie reálnych objektov

Tento piaty krok virtualizačného reťazca uzatvára slučku tým, že umožňuje **transformovať digitálne modely späť do fyzickej podoby**. V kontexte priemyslu 4.0 a digitálnych dvojčiat slúži na rýchlu prototypovú výrobu súčiastok alebo procesov navrhnutých a overených vo virtuálnom svete. Zároveň sem patrí aj využitie technológie **LiDAR** pre 3D skenovanie reálnych objektov a ich následnú digitalizáciu.

36) Vizualizácia, vykresľovanie, riadenie vizualizácie, vizualizačné jadrá a grafické architektúry, obmedzenia vizualizačného podsystému XR.

1. Vizualizácia a vykresľovanie (Rendering)

Zdroje definujú tieto dva pojmy ako komplementárne procesy prepojenia modelu a obrazu:

- **Vizualizácia:** Je proces **transformácie popisu modelu virtuálneho sveta** do výstupného obrazu na zobrazovacom zariadení. V rámci virtualizačného reťazca ide o štvrtý krok (spracovanie a práca s VR), pričom ide o kľúčový podsystém, nakoľko človek spracúva približne **80 % informácií zrakom**.
- **Vykresľovanie (Rendering):** Predstavuje proces **tvorby obrazu pomocou počítačových algoritmov** alebo oblasť štúdia syntézy obrazov z modelov sveta vo výpočtovom systéme. Je to opak k „videniu“ (vision), ktoré obrazy analyzuje na vytvorenie modelov.

2. Vrstvy vizualizačného procesu

Proces vizualizácie v XR systémoch je komplexnejší než v klasickej 3D grafike a delí sa na nasledujúce vrstvy:

1. **Definovanie a spracovanie modelu:** Zahŕňa reprezentáciu a určenie súradnicových systémov.
2. **Geometrické transformácie:** Operácie nad virtuálnymi objektmi.
3. **Riešenie viditeľnosti:** Určenie, ktoré časti scény sú pre pozorovateľa viditeľné.
4. **Tieňovanie a osvetľovanie:** Výpočet interakcie svetla s povrchmi.
5. **Realistické zobrazovanie:** Aplikácia fotorealistických metód a zrkadlenia.
6. **Kompozícia a vykresľovanie:** Rasterizácia a finálne uloženie do zásobníka snímkov.

3. Riadenie vizualizácie v zdieľaných systémoch

V systémoch zdieľanej reality sa uplatňujú dva prístupy k riadeniu zobrazenia:

- **Riadené jedným používateľom:** Typické pre systémy s centralizovaným riadením (napr. virtuálna jaskyňa CAVE). Využíva sa najmä **Server Side Rendering (SSR)**.
- **Riadené každým používateľom:** Využíva sa pri oddelenom prístupe (typicky XR headsety), kde sa uplatňuje **Client Side Rendering (CSR)**.

4. Vizualizačné jadrá a grafické architektúry

Vizualizačné procesné jadro riadi spracovanie dát v troch úrovniach:

- **Úroveň A:** Spracovanie virtuálnych objektov.
- **Úroveň B:** Spracovanie polygónov (transformácie, viditeľnosť, tieňovanie).
- **Úroveň C:** Vykresľovanie a zobrazenie na výstupnom zariadení.

Grafické architektúry majú za úlohu generovať výstup pre zobrazovače a klasifikujú sa podľa použitej **reprezentácie prostredia** a použitého **algoritmu** na tvorbu obrazu. Moderné XR systémy využívajú rozhrania ako **OpenXR**, ktoré nie je rendererom, ale sprostredkúva prepojenie na knižnice (Vulkan, OpenGL, DirectX) a manažuje synchronizáciu a **swapchain** (reťazec bufferov pre plynulé zobrazenie).

5. Obmedzenia vizualizačného podsystemu XR

Vizualizácia v XR čelí špecifickým výzvam, ktoré vyplývajú z potreby prevádzky v reálnom čase:

- **Technické obmedzenia:** Spotreba energie (mobilné XR), teplotné limity HMD a priepustnosť pamäte.
- **Výpočtové obmedzenia:** Rýchlosť zapĺňania GPU (**fill-rate**), zložitosť shaderov a náročnosť komunikácie CPU–GPU.
- **Real-time požiadavky:** Nutnosť vysokej obnovovacej frekvencie (**90–120 Hz**) a nízkej latencie (**motion-to-photon**), aby sa predišlo **cybersickness** a nestabilite.

Dizajnové kompromisy: Návrhár systému musí vyvažovať protichodné požiadavky, ako sú univerzálnosť vs. špecifická optimalizácia, cena vs. výkon, imerzia vs. výkon a kvalita obrazu vs. latencia. Na zmiernenie týchto obmedzení sa využívajú **pokročilé techniky** ako **Foveated rendering** (vyššie rozlíšenie v centre pohľadu) alebo **Time-warp/Space-warp reprojection** (geometrický dopočet snímok pri poklese výkonu).

37) Špecifické požiadavky na vizualizáciu v XR systémoch a pokročilé rendering XR techniky.

Špecifické požiadavky na vizualizáciu v XR

V systéme XR musí vizualizačný podsystem pracovať v **reálnom čase** s cieľom minimalizovať oneskorenie medzi pohybom používateľa a obrazom a zabezpečiť stabilný výstup bez výpadkov. Medzi kľúčové nároky patria:

- **Stereoskopické zobrazovanie:** Spracovanie samostatnej projekcie a renderovanie samostatného pohľadu pre ľavé a pravé oko, čo je základom pre dosiahnutie hĺbkového vnemu.
- **Vysoká obnovovacia frekvencia:** Stabilné udržiavanie frekvencie v rozmedzí **90 – 120 Hz**.
- **Nízka latencia (Motion-to-photon):** Minimalizácia časového intervalu medzi fyzickým pohybom používateľa (napr. hlavy) a momentom, kedy sa tento pohyb prejaví na displeji ako zmenený obrazový pixel.
- **Stabilita snímkovej frekvencie (Frame pacing):** Zabezpečenie rovnomerných časových odstupov medzi jednotlivými snímkami.

Pri implementácii týchto požiadaviek zohráva dôležitú úlohu **OpenXR**, ktoré síce nie je rendererom, ale manažuje synchronizáciu časovania jednotlivých krokov vykresľovania a reťazca obrazových bufferov (**swapchain**).

Pokročilé rendering XR techniky

Tieto techniky slúžia na **zníženie výpočtovej náročnosti** pri zachovaní vysokého komfortu používateľa a plynulosti obrazu:

1. **Foveated rendering:** Využíva vlastnosti ľudského oka a renderuje s vysokým rozlíšením len centrálnu časť pohľadu, pričom v periférnych oblastiach kvalitu obrazu adaptívne znižuje.
2. **Time-warp reprojection:** Ak GPU nestihne včas vyrenderovať nový snímok, táto metóda vezme posledný hotový obraz a geometricky ho prepočíta podľa aktuálnej rotácie hlavy používateľa. Obraz tak reaguje na pohyb okamžite aj pri poklese výkonu.
3. **Space-warp reprojection:** Používa sa pri nízkej snímkovej frekvencii (napr. 30 FPS namiesto 60 FPS). Systém renderuje len každý druhý snímok a medzisnímky umelo dopočítava na základe vektora pohybu objektov v scéne.
4. **Asynchrónne vykresľovanie:** Technika založená na oddelení procesu samotného renderovania od procesu zobrazovania na výstupnom zariadení.

Rozdiely podľa platformy

Spôsob vykresľovania sa líši podľa konštrukcie systému:

- **HMD (tesne-viazané):** Používajú primárne **stereoskopické renderovanie**.
- **Virtuálne jaskyne (voľne-viazané):** Využívajú **viac-pohľadové (multi-view) renderovanie** pre správne zobrazenie na viacerých stenách súčasne.

Cieľom nasadenia týchto pokročilých metód (ako single-pass stereo alebo multi-view) je znížiť výpočtovú záťaž pri zachovaní maximálnej vizuálnej kvality.

38) Paralelné vykresľovanie (paralelný rendering), koncepcie paralelných algoritmov.

Paralelné vykresľovanie (rendering) predstavuje využitie viacerých procesných jednotiek na urýchlenie výpočtu virtuálnej scény. V súčasnosti je táto technológia kľúčovým spôsobom, ako riešiť vysoké nároky na vizualizáciu v XR systémoch.

1. Dôvody nasadenia paralelného renderingu

Hlavnými faktormi, ktoré si vyžadujú paralelizáciu výpočtov, sú:

- **Komplexnosť scény:** Vysoký počet polygónov a objektov.
- **Kvalita výstupného obrazu:** Náročné algoritmy pre tieňovanie, osvetľovanie a realizmus.
- **Interakcia v reálnom čase:** Potreba minimalizovať latenciu a zabezpečiť okamžitú odozvu systému.

2. Koncepcie paralelných algoritmov

V grafických architektúrach sa uplatňujú tri základné koncepcie paralelizmu:

A. Funkcionálny paralelizmus (MISD)

Tento prístup rozdeľuje proces výpočtu na niekoľko jednoznačných **funkcií**, ktoré sa postupne aplikujú na prúd údajov.

- **Princíp:** Ak jedna procesná jednotka zodpovedá jednej funkcii a údaje prúdia z jednej jednotky do druhej, hovoríme o **prúdovom (zreťazenom) výpočte**.
- **Využitie:** Je vhodný pre spracovanie polygónov v rámci tzv. **výpočtovej vizualizačnej rúry** (pipeline), kde na vstupe sú 3D primitíva a na konci hodnoty pixelov.
- **Obmedzenia:** Celková rýchlosť je limitovaná najpomalšou jednotkou v reťazci a počet funkčných jednotiek je konečný.

B. Údajový paralelizmus (SIMD)

Namiesto zreťazenia funkcií sa **údaje rozdelia do viacerých prúdov**, ktoré sú spracovávané rovnocennými procesnými jednotkami súčasne.

- **Triedy údajového paralelizmu:**

- **Objektový paralelizmus:** Zameriava sa na transformačnú fázu (geometrické transformácie, orezanie, výpočty svetiel).
- **Obrazový paralelizmus:** Zameriava sa na rasterizačnú fázu (výpočet osvetlenia pixelov, interpolácia, riešenie viditeľnosti).

- **Závislosti:** Výkon závisí od zložitosti scény, stupňa vzorkovania a rozlíšenia výstupu. Výsledky sa spájajú v zásobníku snímkov (framebuffer).

C. Časový paralelizmus

Využíva sa najmä v animáciách, kde sa problém rozdeľuje v **časovej doméne**.

- **Princíp:** Každý procesor dostane na starosť určitý počet snímok (frames), ktoré vypočíta nezávisle od ostatných.
- **Použitie:** Čas výpočtu jednej snímky tu nie je taký kritický ako celkový čas potrebný na vyrenderovanie celej sekvencie.

3. Hybridný prístup a súvislosti

V moderných systémoch sa najčastejšie využíva **hybridný prístup**, ktorý kombinuje vyššie uvedené koncepcie pre dosiahnutie maximálnej efektivity. Pri návrhu algoritmov sa využívajú aj tzv. **geometrické súvislosti**, ktoré znižujú objem potrebných výpočtov vďaka podobnosti dát:

- **Snímková súvislosť:** Podobnosť pixelov medzi po sebe idúcimi snímkami.
- **Riadková súvislosť:** Podobnosť pixelov v rámci riadkov displeja.
- **Súvislosť span:** Podobnosť pixelov v rámci určitého intervalu v riadku.
- **Lúčová súvislosť:** Využívaná pri algoritmoch sledovania lúča (ray tracing).

39) Paralelizmus v grafických architektúrach.

V systémoch virtuálnej a rozšírenej reality je paralelizmus v grafických architektúrach nevyhnutný na urýchlenie výpočtov zložitej scény, zabezpečenie vysokej kvality obrazu a minimalizáciu latencie pre odozvu v reálnom čase. Rozlišujeme tri základné koncepcie paralelizmu a klasifikáciu architektúr podľa miesta triedenia údajov.

Základné typy paralelizmu

Grafické architektúry využívajú nasledujúce formy paralelizácie výpočtov:

- **Funkcionálny paralelizmus (MISD):** Proces výpočtu sa rozdelí na niekoľko jednoznačných **funkcií**, ktoré sa postupne aplikujú na údaje. Ak jedna procesná jednotka zodpovedá jednej funkcii a dáta prúdia z jednej do druhej, ide o **prúdový (zreťazený) výpočet**. Tento prístup je základom **výpočtovej vizualizačnej rúry**, ktorá zahŕňa kroky ako evaluátor, operácie nad vrcholmi, rasterizácia, textúrovanie, operácie nad fragmentmi a kompozícia.

- **Údajový paralelizmus (SIMD):** Údaje sa rozdelia do viacerých prúdov a pridelia sa **rovnocenným procesným jednotkám**, ktoré pracujú súčasne. Výsledky sa následne spájajú v zásobníku snímok (framebuffer). Delí sa na dve triedy:
 - **Objektový paralelizmus:** Prebieha v transformačnej fáze (geometrické transformácie, orezanie, výpočty svetiel).
 - **Obrazový paralelizmus:** Prebieha v rasterizačnej fáze (výpočet osvetlenia, interpolácia, riešenie viditeľnosti).
- **Časový paralelizmus:** Problém sa rozdeľuje v **časovej doméne**, čo je typické pre animácie. Každý procesor vypočítava určitý počet snímok nezávisle od ostatných. Čas výpočtu jednej snímky tu nie je taký kritický ako celkový čas animácie.

V moderných systémoch sa často využíva **hybridný prístup**, ktorý kombinuje tieto koncepcie pre dosiahnutie maximálneho výkonu.

Klasifikácia grafických architektúr podľa triedenia

Paralelné grafické architektúry sa skladajú z viacerých uzlov (každý má vlastnú vizualizačnú rúru) a klasifikujú sa podľa toho, v ktorej časti rúry nastáva triedenie údajov:

1. **Architektúra s triedením na začiatku (Sort-First):** Príkazová jednotka určí, kam primitívum padne v zásobníku snímok, a pošle ho do príslušného uzla. Využíva **snímkovú súvislosť**, ale problémom môže byť nerovnomerné vyťaženie procesorov.
2. **Architektúra s triedením v strede (Sort-Middle):** Uzly transformujú 3D primitíva do 2D priemetne a následne ich triedia do rasterizačných uzlov podľa rozdelenia obrazovky.
3. **Architektúra s triedením na konci (Sort-Last):** Každý procesor má takmer kompletnú rúru a vytvára časti obrazu. Tie sa spájajú v zásobníku snímok, najčastejšie podľa hĺbky (Z-buffer). Je **výborne škálovateľná** z hľadiska počtu primitív, ale **nevhodná pre vysoké rozlíšenia** kvôli nárokom na prenos pixelov.
4. **Architektúra s kompozíciou obrazu:** Rozdeľuje vypočítané pixely až v procese zobrazovania.
5. **Architektúra s viacnásobným triedením:** Umožňuje komunikáciu "každý s každým" na všetkých úrovniach rúry.

Pri návrhu týchto systémov sa využívajú **geometrické súvislosti** (snímková, riadková, lúčová alebo súvislosť span), ktoré vďaka podobnosti pixelov v priestore alebo čase pomáhajú optimalizovať výpočtovú záťaž.

40) Klasifikácia grafických architektúr a ich porovnanie.

Klasifikácia paralelných grafických architektúr v systémoch XR sa odvíja od miesta vo vizualizačnej rúre, v ktorom nastáva **triedenie údajov**. Tieto architektúry sa zvyčajne skladajú z viacerých uzlov, pričom každý uzol disponuje vlastnou rúrou (evaluátor, operácie nad vrcholmi, rasterizácia, textúrovanie, operácie nad fragmentmi a kompozícia).

Klasifikácia grafických architektúr

Podľa miesta triedenia rozlišujeme päť základných typov:

1. **Architektúra s triedením na začiatku (Sort-First):** Triedi 3D primitíva.
2. **Architektúra s triedením v strede (Sort-Middle):** Triedi 2D primitíva.
3. **Architektúra s triedením na konci (Sort-Last):** Triedi fragmenty.
4. **Architektúra s kompozíciou obrazu:** Rozdeľuje pixely až v procese zobrazovania.
5. **Architektúra s viacnásobným triedením (Multi-sorting):** Umožňuje komunikáciu "každý s každým" na všetkých úrovniach rúry.

Porovnanie hlavných typov architektúr

Vlastnosť	Triedenie na začiatku (Sort-First)	Triedenie v strede (Sort-Middle)	Triedenie na konci (Sort-Last)
Čo sa triedi	3D primitíva	2D primitíva	Fragmenty (časti obrazu)
Princíp práce	Príkazová jednotka určí, kam primitívum padne v obraze, a pošle ho do príslušného uzla na transformáciu a rasterizáciu.	Uzly transformujú 3D primitíva do 2D priemetne a následne ich triedia do rasterizačných uzlov podľa rozdelenia obrazovky.	Každý procesor má takmer kompletnú rúru a vytvára časti obrazu, ktoré sa spájajú (hlbkové triedenie) v zásobníku snímkov.
Výhody	Výrazne využíva snímkovú súvislosť ; pri veľkom počte objektov ich netreba preposielať.	Jasné rozdelenie medzi transformačné a rasterizačné procesory.	Výborná škálovateľnosť počtu primitív – stačí pridať procesory pre spracovanie zložitejšej scény.
Nevýhody	Nerovnomerné vyťaženie systému, ak sú objekty v scéne rozmiestnené neproporčne k regiónom.	Náročnosť na triedenie 2D primitív po transformácii.	Nevhodná pre vysoké rozlíšenia , pretože každý pixel (vrátane farby a hĺbky) sa musí posilať cez spojovaciu sieť.

Výber architektúry závisí od potrieb konkrétneho XR systému. Ak je prioritou spracovanie extrémne vysokého počtu polygónov, výhodná je architektúra s triedením na konci. Ak systém naráža na limity prenosu pixelov pri vysokom rozlíšení, vhodnejšie sú prístupy triediace primitíva skôr. V praxi sa využívajú aj riešenia pre **veľkoplošné zobrazovanie** (napr. systémy ako TileSort), ktoré kombinujú renderovacie uzly pre generovanie viacerých pohľadov súčasne.

41) Technológia GPGPU – základné pojmy, architektúra, rozvoj, prostriedky.

Technológia **GPGPU** predstavuje zásadný posun vo využití grafických kariet, kedy sa ich masívny výpočtový výkon začal využívať na iné než len čisto grafické účely.

1. Základné pojmy

- **Definícia:** GPGPU je skratka pre **General Purpose computing on Graphics Processor Units**, čo znamená vykonávanie všeobecných (negrafických) výpočtov prostredníctvom grafických procesorov.
- **Charakteristika procesora:** GPU je v súčasnosti koncipovaný ako **masívne paralelný prúdový procesor typu SIMD** (Single Instruction, Multiple Data).
- **Výkon:** Súčasné GPU dosahujú výpočtový výkon rádovo na úrovni **TFLOPSov**.

2. Architektúra z pohľadu GPGPU

Architektúra grafického hardvéru pre všeobecné výpočty pozostáva z nasledujúcich kľúčových blokov:

- **Riadiaci procesor (Control processor):** Manažuje beh úloh na strane GPU.
- **Dátovo-paralelné procesorové pole:** Skupina výpočtových jadier určených na paralelný výpočet údajov.
- **Správca pamäte:** Zabezpečuje tok dát (inštrukcie, konštanty, vstupy a výstupy) medzi **Video RAM** a systémovou **RAM**.
- **Komunikácia s CPU:** Aplikácia bežiaca na CPU riadi prenos inštrukcií a dát smerom do GPU zariadenia.

3. Rozvoj a generácie programovateľnosti

Vývoj programovateľnosti GPU prebiehal v piatich generáciách:

1. **1. generácia (1996 – 1997):** Hardvér nebol programovateľný; geometria sa riešila cez CPU a GPU urýchlilo len rasterizáciu (napr. 3dfx Voodoo).
2. **2. generácia (1998 – 1999):** Presun vertex transformácií na GPU; začiatok možností programovania cez 3D API ako OpenGL.
3. **3. generácia (2001 – 2002):** Rozšírené programovanie vertex transformácií a zavedenie jednoduchých **fragment transformácií** pomocou shading jazykov.
4. **4. generácia (2002 – 2006):** Plne programovateľné vertex a fragment transformácie; vznik prvých jednoduchých GPGPU aplikácií a metajazykov.
5. **5. generácia (2007 – súčasnosť):** Zavedenie **unifikovaných vykonávacích jednotiek** (unified shaders), ktoré dokážu podľa potreby vykonávať vertex, fragmentové aj geometrické programy.

4. Prostriedky a platformy

Pre programovanie GPGPU sa v súčasnosti využívajú tri hlavné smery:

- **nVidia CUDA:** Rozšírenie jazyka C vyvinuté spoločnosťou nVIDIA. Využíva hierarchický vývojový model: **vlákno (thread)** → **osnova (warp)** → **blok (block)** → **mriežka (grid)**.
- **OpenCL (Open Computing Language):** Otvorený, agnostický štandard (Khronos Group), ktorý umožňuje programovanie heterogénnych systémov pozostávajúcich z CPU, GPU a iných procesorov (napr. FPGA).

- **DirectX Compute:** Riešenie od spoločnosti Microsoft integrované do API DirectX.

GPGPU technológia nachádza uplatnenie v **umelej inteligencii**, kryptografii, fyzikálnych simuláciách, rozpoznávaní obrazu či spracovaní satelitných dát.

42) Priestorové zobrazovanie - stereoskopické videnie a jeho proces, vlastnosti oka, rozlišovacia schopnosť, zorné pole, akomodácia a konvergencia (VAC), faktory priestorového videnia, Panumova fúzna oblasť

1. Stereoskopické videnie a jeho proces

Stereoskopické videnie je základnou schopnosťou ľudí vnímať svet priestorovo. Ak sa pri zobrazovaní využíva viac než dva pohľady, hovoríme o **multiskopii**. Samotný **proces** pozostáva z troch fáz:

1. **Snímanie:** Realizuje sa pomocou očí.
2. **Spracovanie:** Prebieha v mozgu, kde dochádza k syntéze obrazov a ovládaniu očného svalstva.
3. **Výsledok:** Vzniká hĺbkový vnem, ktorý je individuálny pre každého človeka.

2. Vlastnosti oka a fyziológia

Pre XR systémy sú dôležité najmä tieto časti oka:

- **Rohovka:** Priehľadná vrstva, cez ktorú svetlo vstupuje do oka.
- **Dúhovka a zrenička:** Zrenička reguluje množstvo svetla. Pri jej zmenšení sa zväčšuje **hĺbka ostrosti**. Sledovanie smeru natočenia zreničky (eye tracking) je kľúčové pre mnohé XR aplikácie.
- **Šošovka:** Zabezpečuje zaostrovanie zmenou svojho zakrivenia, čo sa nazýva **akomodácia**.
- **Sietnica:** Obsahuje fotoreceptory (**tyčinky a čapíky**) a dochádza tu k prvému spracovaniu obrazu.

3. Rozlišovacia schopnosť a zorné pole

- **Rozlišovacia schopnosť:** Je to najmenšia uhlová vzdialenosť dvoch bodov, ktoré oko ešte rozozná (monokulárna medza je cca **60"**). Na sietnici to zodpovedá vzdialenosti približne 0,005 mm.
- **Zorné pole:** Priestor zachytený bez zmeny smeru pohľadu. Delí sa na:
 - **Priame videnie:** Ostrá časť vnímaná cez foveu (centrálna časť do 2° – 10°).
 - **Nepriame (periférne) videnie:** Zvyšok poľa (do 30° stredné, 90° – 100° okrajové).
 - **Binokulárny prekryv:** Časť, ktorú vidia obe oči súčasne, má rozsah približne **120°**.

4. Akomodácia, konvergencia a konflikt VAC

- **Konvergencia:** Natočenie očí smerom k pozorovanému objektu.
- **Akomodácia:** Preostrenie šošovky na vzdialenosť objektu.

- **VAC konflikt (Vergence-Accommodation Conflict):** V reálnom svete sú tieto deje viazané. V XR systémoch je obraz fyzicky na fixnej rovine displeja, takže zatiaľ čo konvergencia sa mení podľa virtuálneho objektu, akomodácia ostáva fixná na displej. To spôsobuje únavu očí, bolesti hlavy a nevoľnosť.

5. Faktory priestorového videnia

Hĺbkový vnem ovplyvňujú:

- **2D náznaky (cues):** Relatívna veľkosť, jas, kontrast, tieňovanie a prekrytie objektov.
- **Pohybová paralaxa:** Zmena pohľadu na scénu pri pohybe hlavy.
- **Binokulárna a hĺbková disparita:** Rozdiely medzi obrazmi v ľavom a pravom oku a interpretácia natočenia očí mozgom.
- **Dosah:** Človek priestorovo vníma v rozsahu **0,3 m až 220 m**.

6. Panumova fúzna oblasť (PFO)

PFO je rozsah binokulárnej disparity, v ktorom je mozog ešte schopný zlúčiť obrazy z oboch očí do jedného stabilného vnemu. Ak je disparita mimo túto oblasť, vzniká **diplopia (dvojité videnie)**. Komfortné XR systémy by mali pracovať hlboko pod hranicou PFO. S PFO súvisí aj **horopter** (úroveň nulovej disparity), pričom rozlišujeme skríženú disparitu (objekt pred fixačným bodom) a neskríženú disparitu (objekt za ním).

43) Priestorové zobrazovanie - technológie stereoskopického a priestorového zobrazovania.

Priestorové zobrazovanie predstavuje v XR systémoch technologický a kognitívny **kompromis medzi biológiou videnia a technickými možnosťami**. Existuje široké spektrum technológií, ktoré umožňujú sprostredkovanie hĺbkového vnemu, od tradičných metód až po moderné systémy riešiace biologické limity oka.

Prehľad technológií priestorového zobrazovania

Zdroje uvádzajú nasledujúcu klasifikáciu technológií:

- **2,5D projekcia:** Bežné 2D obrazy, ktoré využívajú len 2D náznaky (cues) na simuláciu hĺbky; neposkytujú verný priestorový vnem.
- **Stereogram:** Využíva binokulárnu disparitu na vytvorenie 3D vnemu z 2D vzorov alebo textúr.
- **Stereoskopia:** Základný princíp zobrazovania odlišného obrazu pre každé oko.
- **Autostereoskopia:** Zobrazovanie 3D obrazu bez potreby okuliarov.
- **Volumetrické zobrazovanie:** Vytváranie skutočného telesa v priestore pomocou rozsvietených bodov.
- **Holografia:** Pokročilé systémy vytvárajúce priestorové svetelné pole.
- **Light-field, varifokálne a multifokálne displeje:** Najmodernejšie systémy zamerané na prirodzenú akomodáciu.

- **Projekcia na očnú sietnicu (retinal projection):** Priame premietanie obrazu do oka.
- **3D tlač:** Využívaná ako hmatový a fyzický priestorový výstup.

Stereoskopické systémy

Sú založené na separácii obrazov pre ľavé a pravé oko a delia sa na dve hlavné skupiny:

1. **Tesne-viazané systémy:** Vyžadujú špeciálne pomôcky na oči, ako sú **anaglyfické okuliare** (farebná separácia R-C alebo Y-B), **polarizačné okuliare** (lineárna, kruhová alebo eliptická polarizácia) alebo **aktívne (shutter) okuliare** synchronizované so zdrojom obrazu frekvenciou 60 – 240 Hz. Patria sem aj moderné **XR headsety (HMD)**.
2. **Voľne-viazané systémy:** Stereoeфекt sa dosahuje úpravou samotného zobrazovacieho zariadenia, čo sa využíva napríklad v systémoch typu **CAVE** (virtuálna jaskyňa).

Špecifickou pasívnou technológiou je **Infitec**, ktorá využíva interferenčné filtre s rôznymi vlnovými dĺžkami pre každé oko, čím umožňuje vnímať plne farebný obraz bez vzájomného rušenia kanálov.

Autostereoskopické technológie

Tieto systémy umožňujú vnímať 3D obraz voľným okom pomocou nasledujúcich metód:

- **Paralaxná bariéra (prizmová mriežka):** Mechanické rozdelenie svetla pre každé oko.
- **Lentikulárny (šošovkový) raster:** Sústava drobných šošoviek usmerňujúcich pixely.
- **Riadenie s aktívnou maskou:** Využíva kamerový systém na sledovanie používateľa a dynamické prispôbenie obrazu.
- **Hĺbková mapa:** Obraz je dopĺňaný informáciou o hĺbke pre každú časť scény (napr. systém Philips WOW).

Pokročilé systémy (Light-field, Multifocal, Varifocal)

Moderné zobrazovacie systémy sa snažia eliminovať **VAC konflikt** (nesúlad medzi akomodáciou a konvergenciou), ktorý spôsobuje únavu v XR.

- **Light-field displeje:** Zobrazujú celé svetelné pole vrátane smeru lúčov, čím umožňujú oku prirodzene zaostrovať.
- **Multifokálne displeje:** Zobrazujú scénu na viacerých diskretných ohniskových rovinách súčasne pomocou vrstvených priehľadných displejov.
- **Varifokálne displeje:** Mechanicky posúvajú šošovky alebo displej podľa toho, kam sa používateľ pozerá, aby zabezpečili ostrý obraz v mieste fokusu.

Volumetrické zobrazovanie a 3D tlač

Volumetrické systémy vytvárajú obraz rozsvietením bodov v skutočnom fyzickom priestore, napríklad pomocou rotujúcej dosky alebo LED kocky. Ich výhodou je, že nespôsobujú nevoľnosť, nevýhodou však ostáva vysoká cena a konštrukčná náročnosť. **3D tlač** uzatvára reťazec tým, že umožňuje digitálny model z virtuálneho sveta premeniť na fyzický objekt, čo má veľký význam napríklad pre nevidiacich používateľov.

44) Stereogramy.

1. Princíp fungovania stereogramu

Stereogramy (vrátane stereopárov) využívajú **binokulárnu disparitu** v kombinácii so schopnosťou mozgu detegovať aj veľmi malé rozdiely medzi obrazmi videnými ľavým a pravým okom. Na základe týchto rozdielov mozog spätne vygeneruje **ilúziu 3D povrchu**.

2. Kľúčové techniky a typy

V rámci vývoja stereogramov zdroje uvádzajú niekoľko významných míľnikov a technológií:

- **Random Dot Stereogram (stereogram náhodných bodiek):** Najzákladnejšia technika vyvinutá **Bélom Juleszom** v roku 1959.
- **Autostereogram (Single Image Stereogram):** Technika, ktorú v roku 1979 predstavil **Christopher Tyler**. Umožňuje vnímať 3D vnem z jedného obrázka bez špeciálnych pomôcok.
- **2,5D projekcia:** Hoci sa niekedy spomína v kontexte 3D, rešpektuje iba 2D náznaky (cues) a neposkytuje verný priestorový vnem.

3. Proces generovania a vizualizácie

Pre správne technické vytvorenie stereogramu (najmä typu SIS) je nevyhnutná **hlĺbková mapa obrazu**, prípadne mapa skrytých hrán alebo povrchov. Samotné stereogramy majú dôležitú vlastnosť – sú **invariantné voči zmene mierky**.

Aby používateľ uvidiel skrytý 3D objekt, musí zmeniť spôsob, akým sa jeho oči zameriavajú na obraz. Rozlišujeme dva základné spôsoby vizualizácie:

- **Divergentná vizualizácia (pohľad):** Osy pohľadu očí smerujú akoby "za" rovinu obrázka.
- **Konvergentná vizualizácia (krížový pohľad):** Osy pohľadu sa krížia v bode pred obrázkom.

4. Historické nadväznosti

Koncepcia stereogramov úzko súvisí s prácou priekopníkov v oblasti priestorového videnia:

- **Charles Wheatstone (1838):** Prvé vedecké popisy stereovidenia a vynález zrkadlového stereoskopu.
- **David Brewster (1849):** Zdokonalenie šošovkového stereoskopu.

V praxi sa princípy stereogramov využívajú pri mapovaní rozličných textúr s rozdielnym hlĺbkovým obrazom na vytvorenie komplexných vizuálnych ilúzií.

45) Stereoskopia – anaglyfické zobrazenie.

1. Princíp a definícia anaglyfu

Anaglyf je technika založená na **farebnej separácii v RGB farebnom priestore**, kedy je obraz pre každé oko filtrovaný do inej vlnovej dĺžky (farby). Používateľ sleduje výsledný zložený obraz cez **anaglyfické okuliare** s farebnými filtermi, ktoré zabezpečia, že každé oko uvidí len jemu určenú perspektívu, čím v mozgu vzniká hlĺbkový vnem.

2. Klasifikácia anaglyfického zobrazenia

Zdroje rozdeľujú anaglyfy podľa dvoch hlavných kritérií:

- **Podľa farebnosti výstupu:**
 - **V odtieňoch šedej:** Využíva len separované farebné vrstvy na sprostredkovanie hĺbkového vnemu bez pôvodnej farby scény.
 - **Farebné:** Pokúša sa zachovať farebnú informáciu objektov v scéne.
- **Podľa použitých farebných zložiek (kombinácie filtrov):**
 - **R-C (Red-Cyan):** Najbežnejší typ (červená – azúrová).
 - **R-G (Red-Green):** Červená – zelená.
 - **R-B (Red-Blue):** Červená – modrá.
 - **Y-B (Yellow-Blue / ColorCode®):** Žltá-modrá kombinácia, známa pod komerčným názvom ColorCode.

3. Proces skladania vrstiev

Vytvorenie anaglyfu prebieha procesom **skladania vrstiev**, kedy sa vezme obraz z ľavej kamery (L) a pravej kamery (P), pričom každému sa priradí zodpovedajúci farebný filter podľa typu okuliarov a následne sa zlúčia do jedného finálneho obrazu.

4. Výhody a nevýhody

Výhody:

- **Nízke náklady:** Na zobrazenie nie je potrebný špeciálny hardvér; stačia papierové okuliare a bežný displej.
- **Univerzálnosť:** Možnosť použitia v elektronických (monitor, TV, smartfón) aj tlačенých zdrojoch (printové dokumenty).
- **Kompatibilita:** Funguje so štandardnými monitormi, projektormi a projekčnými plátnami.

Nevýhody:

- **Redukcia farieb:** Výrazne obmedzuje a skresľuje viditeľný farebný priestor.
- **Zdravotný komfort:** Používanie môže spôsobovať **bolesti očí, hlavy, nevoľnosť a závraty**.
- **VAC konflikt:** Systém nerešpektuje prirodzenú súvislosť medzi akomodáciou, konvergenciou a vzdialenosťou predmetu, čo prispieva k únave používateľa.

Anaglyf sa v rámci stereoskopických zobrazovacích systémov radí medzi **pasívne techniky**. Hoci je v profesionálnej XR oblasti dnes už prekonaný modernými headsetmi, stále sa využíva pre svoju extrémnu dostupnosť.

46) Stereoskopia – polarizácia, infitec, autostereoskopia.

1. Polarizácia

Polarizácia predstavuje usmernenie vzájomnej polohy elektrických a magnetických polí elektromagnetického vlnenia vo viditeľnom spektre svetla. V rámci stereoskopie ide o kľúčovú technológiu pre **pasívne systémy**.

- **Princíp:** Vyžaduje nasadenie adekvátnych polarizačných filtrov na strane projektorov aj na strane okuliarov používateľa.
- **Typy polarizácie:** V XR systémoch sa najčastejšie využíva **lineárna** (horizontálna/vertikálna), **kruhová** (cirkulárna) alebo **eliptická** polarizácia.
- **Využitie:** Táto technológia je populárna v systémoch typu **CAVE** (virtuálna jaskyňa), kde umožňuje viacerým používateľom sledovať priestorový obraz súčasne.

2. Infitec

Infitec (skratka pre *INterference Filter TEChnology*) je založený na technológii **vlnového multiplexu (WMT)**.

- **Analógia s anaglyfom:** Funguje na podobnom princípe farebnej separácie (RGB), ale s oveľa vyššou presnosťou.
- **Mechanizmus:** Využíva interferenčné filtre so strmými hranami, ktoré rozdelia RGB spektrum na špecifické pásma pre ľavé a pravé oko tak, aby obe oči vnímali **plne farebný obraz** bez vzájomného rušenia kanálov.
- **Klasifikácia:** Radí sa medzi pasívne stereoskopické technológie a vyžaduje filtre na projektoroch aj okuliaroch. Výhodou je kompatibilita s bežnými projekčnými plochami, ktoré nemusia zachovávať polarizáciu.

3. Autostereoskopia

Autostereoskopia predstavuje technológiu zobrazovania priestorového obrazu **bez potreby akýchkoľvek pomôcok** na očiach používateľa (bez okuliarov či headsetov).

- **Historické nadväznosti:** Prvé práce v tejto oblasti siahajú do roku 1940 k výskumu D. V. Surenského a S. P. Ivanova.
- **Používané technológie:**
 - **Paralaxná bariéra (prizmová mriežka):** Mechanické rozdelenie svetla pre každé oko pomocou štrbín.
 - **Lentikulárny (šošovkový) raster:** Sústava drobných šošoviek usmerňujúcich pixely do správneho oka.
 - **Riadenie hĺbkovou mapou:** Pridanie informácie o hĺbke ku klasickému 2D obrazu (napr. systém Philips WOW).
 - **Riadenie s aktívnou maskou:** Moderný prístup využívajúci **kamerový systém** na sledovanie polohy očí používateľa, čím sa rozširuje pozorovací uhol na cca $\pm 25^\circ$.

- **Výhody a limity:** Hlavnou výhodou je vysoký komfort (žiadne okuliare), nevýhodou však ostáva obmedzenie na nízky počet pozorovateľov v optimálnej polohe.

Zobrazovanie v XR systémoch dnes smeruje k ďalšej evolúcii týchto princípov vo forme **light-field**, varifokálnych a multifokálnych displejov, ktoré sa snažia o ešte prirodzenejší vnem a riešenie biologických konfliktov oka.

47) Priestorové zobrazovanie – hĺbkový vnem, problémy zobrazovania v HMD, stereoskopická súradnicová sústava, virtuálne kamery.

1. Hĺbkový vnem

Hĺbkový vnem (priestorové vnímanie) je schopnosť mozgu interpretovať vzdialenosť objektov.

- **Rozsah vnímania:** Človek priestorovo vníma v rozsahu vzdialeností $D \approx 0,3 \text{ m}$ až 220 m .
- **Stereoskopické rozlíšenie:** Tento vnem je daný vzdialenosťou očí ($B \approx 65 \text{ mm}$) a uhlom α . Vzťah je definovaný rovnicou $D = \frac{B}{2 \cdot \tan(\alpha/2)}$.
- **Vplyv pohybu:** So zvyšujúcou sa rýchlosťou používateľa voči okoliu sa interval priestorového vnímania D skracuje.
- **Faktory ovplyvňujúce vnem:** Okrem binokulárnej disparity (rozdiel obrazov v očiach) sem patria **2D náznaky** (veľkosť, kontrast, tieňovanie), **pohybová paralaxa** (zmena pohľadu pri pohybe) a **hĺbková disparita**, kedy mozog interpretuje natočenie očí (konvergenciu).

2. Problémy zobrazovania v HMD (Konflikt VAC)

Hlavným obmedzením súčasných XR headsetov je **konflikt konvergencie a akomodácie (VAC – Vergence-Accommodation Conflict)**.

- **Podstata problému:** V reálnom svete sú tieto dva javy viazané. V HMD je však obraz fyzicky v jednej fixnej rovine displeja, takže zatiaľ čo konvergencia (natočenie očí) sa mení podľa virtuálneho objektu, akomodácia (zaostrenie šošovky) zostáva fixná na displej.
- **Dôsledky:** Tento nesúlad spôsobuje **únavu očí, bolesti hlavy a nevoľnosť** (cybersickness), ktorá často pramení z nestability vnímania v súvislosti s Panumovou fúznou oblasťou. Zároveň to vedie k obmedzeniu času, počas ktorého môže používateľ systém komfortne používať.

3. Stereoskopická súradnicová sústava

Pre správne vygenerovanie priestorového obrazu systém využíva komplexnú sústavu súradníc, ktorá prepája virtuálny svet s očami používateľa. Zahŕňa:

- **USS (Univerzálna súradnicová sústava):** Globálny priestor scény.
- **SSCL a SSCP:** Lokálne súradnicové sústavy pre ľavú a pravú kameru (oko).
- **SSZL a SSZP:** Súradnicové sústavy zobrazovačov (displejov).
- **Základňa (d):** Vzdialenosť medzi virtuálnymi kamerami, ktorá musí zodpovedať anatómii používateľa.

4. Virtuálne kamery a princíp zobrazenia

Nastavenie virtuálnych kamier v XR systéme vyžaduje presnú matematickú kalibráciu parametrov.

- **Kľúčové parametre:**
 - **f:** Ohnisková vzdialenosť kamery.
 - **2t:** Stereoskopická základňa pre získanie obrazov.
 - **C:** Vzdialenosť najbližšej zobrazovanej roviny.
 - **e:** Lineárna paralaxa nekonečna.
- **Výpočtové vzťahy:** Poloha kamier sa určuje vzťahom $t = \{C \times e\} / \{2 \times f\}$. Ohnisková vzdialenosť (focus) sa následne odvodzuje z uhla zorného poľa Φ a šírky obrazu ako:
$$\text{focus} = \{\text{halfx}\} / \{\tan(\Phi/2)\}.$$

Cieľom správneho nastavenia virtuálnych kamier a riešenia optických konfliktov je dosiahnuť stabilný stereoskopický výstup, ktorý minimalizuje zdravotné riziká pre používateľa.

48) Stereoskopické zobrazovacie systémy.

Stereoskopické zobrazovacie systémy tvoria kľúčovú súčasť vizualizačného podsystemu v XR a sú založené na princípe **zobrazovania iného obrázka do každého oka**. Tieto systémy sa klasifikujú podľa viacerých technických a konštrukčných kritérií.

1. Klasifikácia podľa spôsobu vytvárania stereoefektu

Základné rozdelenie stereoskopických systémov je na pasívne, aktívne a autostereoskopické:

- **Pasívne systémy:** Nevyžadujú napájanie okuliarov. K separácii obrazov využívajú fyzikálne vlastnosti svetla alebo farebné filtre.
 - **Anaglyf:** Najstaršia metóda využívajúca farebnú separáciu v RGB priestore (napr. R-C, Y-B).
 - **Polarizácia:** Využíva usmernenie elektromagnetického vlnenia. Najčastejšie ide o lineárnu, kruhovú (cirkulárnu) alebo eliptickú polarizáciu.
 - **Infitec:** Technológia vlnového multiplexu (WMT), ktorá rozdeľuje RGB spektrum na špecifické pásma pre každé oko pomocou interferenčných filtrov, čo umožňuje vidieť plne farebný obraz.
- **Aktívne systémy:** Využívajú synchronizované (shutter) okuliare, ktoré striedavo zatmievajú ľavé a pravé oko v súlade so zobrazovaným obsahom.
 - Pracujú na frekvenciách **60 – 240 Hz**.
 - Vyžadujú **vysielač synchrosignálu** (napr. cez IrDa, RF, Bluetooth, USB).

- **Autostereoskopické systémy:** Umožňujú vnímať 3D obraz **bez akýchkoľvek pomôcok** na očiach.
 - Využívajú technológie ako **paralaxná bariéra** (prizmová mriežka), **lentikulárny raster**, Fresnelove šošovky alebo riadenie pomocou aktívnej masky s kamerovým systémom na sledovanie očí.

2. Klasifikácia podľa konštrukcie a väzby

- **Tesne-viazané systémy:** Stereoeфекt sa dosahuje cez zariadenie priamo na tele používateľa, ako sú **XR headsety (HMD)**, stereoskopy alebo špeciálne okuliare.
- **Voľne-viazané systémy:** Stereoeфекt zabezpečuje úprava samotného zobrazovacieho zariadenia, čo je typické pre **virtuálne jaskyne (CAVE)**.

3. Klasifikácia podľa umiestnenia zdroja

Pri projekčných systémoch (najmä vo virtuálnych jaskyniach) rozlišujeme:

- **Priama (predná) projekcia:** Zdroj obrazu (projektor) je pred plátnom z pohľadu používateľa.
- **Spätná projekcia:** Projektor je umiestnený za polopriehľadným plátnom, čo eliminuje vrhanie tieňov používateľom v scéne.

4. Aplikačné príklady

- **Monitory a TV:** Môžu využívať všetky tri typy (pasívne, aktívne aj autostereoskopické).
- **Veľkoplošné systémy:** Typicky využívajú pasívnu cirkulárnu polarizáciu alebo aktívne systémy pre viacerých pozorovateľov súčasne.
- **Budúcnosť zobrazovania:** Smeruje k **light-field**, **varifokálnym** a **multifokálnym** displejom, ktoré riešia konflikt akomodácie a konvergenzie (VAC), čím zvyšujú komfort v XR systémoch.

49) Kolízie vo virtuálnych scénach, typy kolíznych senzorov, modely kolíznych objektov, dôvody riešenia kolízií.

1. Definícia kolízie vo virtuálnej scéne

Kolízia je stav alebo udalosť, pri ktorej sa dve alebo viaceré entity (objekt, avatar, kamera a pod.) vo virtuálnom alebo zmiešanom priestore „**fyzicky**“ **dotknú, prekryjú alebo do seba narazia**. V danom momente entity (alebo ich zjednodušené kolízne útvary) obsadzujú rovnakú časť priestoru, čo predstavuje neželaný alebo hraničný stav, ktorý systém musí spracovať.

2. Kolajder (Kolízny objekt)

Kolajder (*collider*) je spravidla neviditeľný, geometricky zjednodušený útvar slúžiaci na detekciu kolízií. Určuje a ohraničuje fyzikálny tvar objektu a zachytáva interakcie s inými kolajdermi.

Najčastejšie typy kolajderov:

- **Guľa (Sphere Collider):** Najjednoduchší útvar, ktorého výpočet je najrýchlejší.
- **Kváder (Box Collider):** Umožňuje najefektívnejšie delenie priestoru.

- **Tesná konvexná obálka (Mesh Collider):** Najpresnejší typ, ktorý presne kopíruje povrch, ale je výpočtovo veľmi náročný.
- **Kapsula (Capsule Collider):** Najčastejšie využívaný kolajder pre postavy a avatarov.

3. Typy kolíznych senzorov (Triggerov)

Kolízne senzory slúžia na aktiváciu dejov v systéme na základe polohy alebo interakcie. Zdroje ich rozdeľujú na:

- **Vertikálny senzorový polygón:** Určený na detekciu z pohľadu zobrazovača; aktivuje sa, ak používateľ ukáže na objekt zariadením (napr. myšou alebo dátovou rukavicou) a stlačí aktivačný prvok.
- **Horizontálny senzorový polygón:** Funguje ako **polohový senzor**, ktorý sa aktivuje automaticky pri vstupe kamery (pozorovateľa) do určitej oblasti.
- **Pohľadový senzor:** Detekuje **ihlan pohľadu kamery** a aktivuje sa, keď sa objekt dostane do zorného poľa. Pri tomto type je nutné detekovať najbližší objekt k osi pohľadu.

4. Modely kolíznych objektov

Pre potreby výpočtov sa objekty rozdeľujú podľa ich matematického popisu, pričom pre každý typ existuje iná forma detekcie:

- **Polygonálne:**
 - **Štruktúrované:** Konvexné alebo konkávne polygóny.
 - **Neštruktúrované:** Polygóny bez informácie o typológii modelu (nie sú pospájané).
- **Nepolygonálne:** Objekty definované funkciami, implicitnými plochami alebo parametrickými plochami.

5. Dôvody riešenia kolízií

Riešenie kolízií je nevyhnutné pre správne fungovanie simulácie a komfort používateľa:

- **Zastavenie pohybu alebo zmena trajektórie:** Aby objekty neprechádzali jeden cez druhý.
- **Zamedzenie prechodu kamery cez steny:** Aby sa pozorovateľ „neprepadol“ cez objekty scény.
- **Využitie v senzorických systémoch:** Napríklad na vykreslenie doplnkových informácií na obrazovku po vstupe do definovanej oblasti.
- **Fyzikálny realizmus:** Umožňuje simulovať odrazy, deformácie a animačné efekty podľa fyzikálnych zákonov.

50) Detekcia kolízie, algoritmy detekcie kolízie, fázy detekcie kolízie.

1. Detekcia kolízie (Collision Detection)

Detekcia kolízie je proces, ktorý v systéme XR rieši fundamentálnu otázku: „Došlo ku kolízii?“.

Výsledkom tohto procesu je:

- **Indikácia** vzniku kolízie.
- Určenie **formy** kolízie a identifikácia zasiahnutých **objektov**.
- Zistenie presného **času**, kedy ku kolízii došlo.

Po úspešnej detekcii spravidla nasleduje **reakcia na kolíziu** (*collision response*), ktorá určuje správanie objektov (zastavenie, odraz, deformácia).

2. Postup detekcie kolízie

Zdroje definujú štandardný procesný postup detekcie v piatich krokoch:

1. **Výber objektov** určených na testovanie.
2. **Utriedenie objektov** a voľba vhodných metód a algoritmov.
3. Aplikácia algoritmov a **test vylúčenia** (eliminácia dvojíc objektov, ktoré sa určite nedotýkajú).
4. Samotný **výpočet kolízie** medzi zostávajúcimi objektmi.
5. **Indikácia** detekcie kolízie pre logické jadro systému.

3. Fázy detekcie kolízie

Vo fyzikálnych a herných enginoch sa na optimalizáciu výkonu využíva **dvojfázová architektúra**:

A. Hrubá fáza (Broad Phase)

Jej cieľom je **výrazne znížiť počet testovaných dvojíc** pomocou rýchlych vylučovacích testov.

- Pracuje s aproximáciami a zjednodušenými kolajdermi.
- Využíva metódy ako **BSH** (hierarchia gúľ), **AABB/OBB** hierarchie alebo **priestorové delenie** (napr. BSP stromy či OCTREE).

B. Jemná fáza (Narrow Phase)

Prebieha len nad objektmi, ktoré prešli hrubou fázou, a vykonáva **presnú geometrickú detekciu**.

- Využíva **konvexné obálky (CHs)** a presné priesečníkové testy nad polygónmi.

4. Algoritmy detekcie kolízie

Algoritmy sa klasifikujú podľa viacerých kritérií:

- **Podľa reprezentácie výstupu:**
 - **Detekčné:** Vracia len logickú hodnotu (pravda/nepravda).
 - **Určujúce:** Presne definuje miesto kolízie (priesečníky, kolízne čiary).
- **Podľa počtu objektov:**
 - **2-Body system:** Rieši kolíziu len dvoch konkrétnych objektov.
 - **N-Body system:** Vypočítava kolízie medzi všetkými pohybujúcimi sa objektmi naraz a hľadá čas stretu pri známych rýchlostiach.

- **Kontinuálna detekcia (CCD - Continuous Collision Detection):**

- Sleduje pohyb objektu plynule v čase na určenie presného okamihu nárazu (*time of impact*).
- Využíva metódu **swept volume** (objem vytvorený pohybom telesa po trajektórii).
- CCD rieši kritický problém **tunnelingu**, kedy rýchlo sa pohybujúci objekt pri diskkrétnej detekcii „preskočí“ cez iný objekt bez zaznamenania kolízie.

5. Optimalizácia pomocou GPU

Pre rozsiahle scény s veľkým počtom objektov sa využíva **GPU akcelerácia**. Grafický procesor sa vďaka masívnemu paralelismu využíva najmä pre **hrubú fázu detekcie**, kedy sa priestor rozdelí na pravidelnú mriežku (*grid-based*) a objekty sa spracúvajú paralelne.

51) Algoritmy riešenia kolízií vo virtuálnych scénach.

1. Podstata riešenia kolízií

Riešenie kolízií je kritickou súčasťou logického jadra systému a fyzikálnych modulov v rámci tzv. **XR rúry (pipeline)**. Jeho cieľom je zabezpečiť realistické správanie virtuálneho prostredia a zabrániť nežiaducim stavom, ako je prechádzanie objektov jeden cez druhý.

2. Kľúčové ciele a dôvody riešenia

Algoritmy riešenia kolízií sa implementujú z nasledujúcich dôvodov:

- **Zastavenie pohybu:** Okamžité prerušenie translácie objektu pri kontakte s prekážkou.
- **Zmena trajektórie pohybu:** Odraz objektu alebo zmena jeho smeru na základe fyzikálnych parametrov.
- **Zamedzenie prieniku kamery:** Zabezpečenie, aby pozorovateľ (kamera/avatar) neprechádzal cez steny a pevné objekty scény.
- **Aktivácia senzorov:** Vyvolanie udalosti (napr. zobrazenie informácií) po vstupe do definovanej kolíznej oblasti.

3. Typy reakcií na kolíziu

Podľa komplexnosti simulácie delíme reakcie do niekoľkých kategórií:

- **Správanie podľa fyzikálnych zákonov:** Výpočet nových síl, rýchlostí a hybnosti po náraze. Využívajú sa tu nástroje ako **PhysX** (pre rigidné telesá a základnú dynamiku) alebo **Ammo.js** (pre webové platformy).
- **Deformácia objektov:** Zmena geometrie kolidujúcich telies (typické pre pružné alebo mäkké objekty). Informácie o deformácii sú spravidla súčasťou **kinematického a statokineticého rámca** objektu.
- **Animačné efekty:** Spustenie preddefinovanej vizuálnej alebo akustickej sekvencie (napr. rozbitie skla, zvuk pri kolízii). Zvukové charakteristiky nárazu sú definované v **akustickom rámci** objektu.

4. Implementácia v XR systémoch

V moderných systémoch sú tieto algoritmy mapované na **logické/procesné jadro (XR core)** a úzko spolupracujú s transformačným reťazcom. Pri modelovaní organizmov sa riešenie kolízií kombinuje s **fyzikálnym a kinematickým modelom**, čím sa dosahuje vysoká miera realizmu (imerzie) pri zachovaní potrebnej stability a výkonu systému.

52) Modely organizmov, skinning, emocionálny model

1. Typy modelov organizmov

V systémoch XR sa organizmy nereprezentujú len ako jednoduché vizuálne objekty, ale ako komplexné štruktúry zložené z viacerých vrstiev. Zdroje uvádzajú nasledujúce typy modelov:

- **Skeletálny model:** Predstavuje kostru a definuje hierarchiu kostí.
- **Muskulárny model:** Reprezentuje svalstvo a slúži najmä na detekciu kolízií a ohraničenie pohybov.
- **Povrchový model:** Predstavuje kožu a finálnu vizualizáciu objektu.
- **Fyzikálny model:** Zabezpečuje interakciu s fyzikálnymi silami v prostredí.
- **Riadiaci model:** Zodpovedá za správanie a umelú inteligenciu (AI).
- **Emocionálny model:** Umožňuje vyjadrovať nálady a pocity.
- **Kinematický (pohybový) model:** Definuje matematický popis pohybov.

Moderné XR systémy využívajú **hybridné prístupy**, kde kombinácia týchto čiastkových modelov tvorí jeden funkčný celok.

2. Kombinácie modelov a ECS

Na správu týchto vrstiev sa využíva koncept **Entity-Component System (ECS)**.

- **Derivované typy:** Umožňujú vytvárať rôzne postavy kombinovaním rôznych sád parametrov (napr. konkrétna kostra + konkrétne svalstvo + povrch).
- **Výhody hybridizácie:** Medzi hlavné prínosy patrí možnosť vytvorenia veľkého množstva unikátnych objektov z obmedzenej sady základov, deľba práce medzi špecialistov a možnosť škálovať náročnosť modelov podľa výkonu platformy.

3. Skinning (Prepojenie kostry a povrchu)

Skinning je kľúčový proces, ktorý určuje, ako sa povrchový model (mesh) deformuje pri pohybe kostí. Je to technický most medzi kinematikou a vizualizáciou.

- **Princíp:** Každý vrchol povrchového modelu je viazaný na jednu alebo viac kostí a má priradené tzv. **váhy (vertex weights)**, ktoré určujú mieru vplyvu kosti na daný bod.

- **Základné techniky:**
 - **Linear Blend Skinning (LBS):** Rýchla a jednoduchá technika, kde sa poloha vrcholu počíta ako vážený lineárny priemer transformačných matíc. Nevýhodou je možný „kolaps kĺbov“ (neprirodzené zužovanie v ohyboch).
 - **Dual Quaternion Skinning (DQS):** Využíva **duálne kvaternióny** pre plynulé transformácie. Lepšie zachováva objem modelu a vytvára realistickejšie deformácie, no je výpočtovo náročnejšia.

4. Emocionálny model a mimika

Emocionálny model v XR úzko súvisí s **tvárovou animáciou (facial animation)** a mimikou. Na jeho realizáciu sa využíva technika **prelínania (morfovania) tvarov** (Blend Shapes alebo Morph Targets).

- **Funkcia:** Táto metóda interpoluje medzi viacerými preddefinovanými tvarmi jedného modelu.
- **Využitie:** Umožňuje jemné svalové pohyby a nekinematické deformácie, ktoré sú nevyhnutné pre vierohodné vyjadrenie emócií.
- **Implementácia:** Prelínanie tvarov sa spravidla kombinuje so skinningom, pričom rieši detaily (pohyby tváre), ktoré samotná kinematika kostry nedokáže efektívne popísať.

53) Pohyby modelov, typy kĺbov, tvorba kinematického reťazca.

1. Pohyby modelov a kinematický reťazec

Pohyb vo virtuálnom priestore je definovaný ako **zmena polohy objektu voči inému objektu**, ideálne v rámci konkrétnej súradnicovej sústavy. Základom pre pohyb zložitejších modelov (ako sú ľudia alebo zvieratá) je kĺbová štruktúra.

Zloženie kinematického reťazca:

- **Báza (statický kĺb):** Pevný bod, voči ktorému sa pohyb vzťahuje.
- **Koncový efektor (dynamický kĺb):** Bod, ktorého zmenu polohy v priestore sledujeme.

Tieto dva prvky spolu tvoria **kinematický reťazec**, ktorého dĺžka môže byť ľubovoľná. Pohyb je realizovaný predovšetkým rotáciou kĺbov, pričom počet osí rotácie určuje **stupeň voľnosti kĺbu (DOF)**. Bežný kĺb má zvyčajne **tri stupne voľnosti**.

2. Typy kĺbov

Z hľadiska základnej mechaniky zdroje rozlišujú dva hlavné typy:

- **Translačný kĺb:** Umožňuje posuvný pohyb.
- **Rotačný kĺb:** Umožňuje otáčavý pohyb.

Pri detailnom modelovaní, napríklad **ľudskej ruky**, sa využívajú špecifické typy kĺbov podľa ich anatomickej funkcie:

- **Metakarpálny falangeálny kĺb (MCP):** Kĺb v spodnej časti prsta, označovaný ako **kladkový typ**.

- **Proximálny interfalangeálny kĺb (PIP):** Kĺb v strede prsta, označovaný ako **guľový typ**.
- **Distálny interfalangeálny kĺb (DIP):** Kĺb najbližšie ku končekom prstov, označovaný ako **sedlový typ**.

3. Tvorba kinematického reťazca

Abstrakcia kĺbovej štruktúry sa v XR systémoch reprezentuje pomocou **acyklického grafu**, konkrétne ako **nepravidelný n-árny strom**.

- **Uzly stromu** reprezentujú samotné kĺby.
- **Spojnice (hrany)** medzi uzlami reprezentujú kosti kĺbov.

Pri hľadaní a definovaní konkrétneho kinematického reťazca v tejto hierarchii môžu nastať **tri základné variácie** vzťahu medzi bázou a efektorom:

1. **Koncový efektor je potomkom bázy:** Štandardný hierarchický postup od stredu tela ku končatine.
2. **Báza je potomkom koncového efektora:** Opačný postup pri výpočtoch.
3. **Efektor a báza sa nachádzajú v rozdielnych podstromoch:** Vyžaduje zložitejšie prechádzanie grafom cez spoločného predka.

Táto hierarchická štruktúra je nevyhnutná pre aplikáciu algoritmov **priamej kinematiky** (výpočet polohy z uhlov) a **inverznej kinematiky** (výpočet uhlov z cieľovej polohy).

54) Pohyby modelov, kĺbová štruktúra, priama kinematika.

1. Kĺbová štruktúra a jej abstrakcia

Pohyb zložitejších modelov (ako sú ľudia alebo zvieratá) je v XR systémoch realizovaný pomocou kĺbových štruktúr.

- **Formálna reprezentácia:** Štruktúra kĺbov sa abstrahuje ako **acyklický graf**, konkrétne ako **nepravidelný n-árny strom**.
 - **Uzly stromu** predstavujú samotné **kĺby**.
 - **Spojnice (hrany)** medzi uzlami predstavujú **kosti**.
- **Typy kĺbov:**
 - **Základné: Translačný** (posuvný) a **rotačný** (otáčavý).
 - **Anatomické (príklad ruky): MCP** (metakarpálny falangeálny – kladkový typ), **PIP** (proximálny interfalangeálny – guľový typ) a **DIP** (distálny interfalangeálny – sedlový typ).

2. Kinematický reťazec

Kinematický reťazec je postupnosť kĺbov a kostí, v ktorej rozlišujeme dva kľúčové body:

- **Báza (statický kĺb):** Pevný počiatočný bod, voči ktorému je pohyb definovaný.

- **Koncový efektor (dynamický kĺb):** Bod na konci reťazca, ktorého výslednú zmenu polohy v priestore sledujeme.
- **Stupeň voľnosti (DOF):** Počet nezávislých smerov (translačných alebo rotačných), v ktorých sa môže kĺb pohybovať. Bežný kĺb má spravidla **3 stupne voľnosti**.

3. Priama kinematika (PK / Forward Kinematics)

Priama kinematika je proces, pri ktorom **vypočítavame polohu koncového efektora na základe známych uhlov kĺbov**.

- **Základná rovnica:** $f(\text{uhly_kĺbov}) = \text{poloha_koncového_efektora}$.
- **Mechanizmus výpočtu:** Výsledná poloha sa určuje postupným násobením transformačných matic (alebo pomocou **quaterniónov**) v rámci kinematického reťazca od bázy smerom k efektoru.
- **Trajektórie:** Pri pohybe rozlišujeme trajektóriu koncového bodu (C_{AB}) a trajektórie jednotlivých kĺbov ($C_{K1}, C_{K2}, \dots$), ktoré sú od seba závislé.

4. Vlastnosti výpočtu (Komutatívnosť)

Zdroje zdôrazňujú dôležitý rozdiel v komutatívnosti procesov priamej kinematiky:

- **Z hľadiska koncového bodu:** Výpočet **je komutatívny**. Je jedno, v akom poradí transformácie aplikujeme, výsledná poloha efektora v cieľovom stave bude rovnaká.
- **Z hľadiska trajektórie:** Výpočet **nie je komutatívny**. Rôzne poradie aplikácie transformácií (napr. najprv rotácia v ramene, potom v lakti vs. naopak) vytvorí **rôzne dráhy pohybu** koncového bodu aj kĺbov.

Tento model je základom pre vizualizáciu a animáciu v reálnom čase, pričom sa často kombinuje s technikou **skinningu**, ktorá prenáša matematický pohyb kostry na vizuálny povrch modelu.

55) Pohyby modelov, kĺbová štruktúra, inverzná kinematika.

1. Pohyby modelov a kinematický reťazec

Pohyb vo virtuálnom priestore je definovaný ako **zmena polohy objektu voči inému objektu**, ideálne v rámci konkrétnej súradnicovej sústavy. Pre simuláciu pohybu zložitejších telies (napr. humanoidov) sa využíva kĺbová štruktúra, v ktorej rozlišujeme:

- **Báza (statický kĺb):** Pevný bod, voči ktorému sa pohyb reťazca vzťahuje.
- **Koncový efektor (dynamický kĺb):** Bod na konci reťazca, ktorého zmenu polohy v priestore sledujeme.
- **Stupeň voľnosti (DOF):** Definuje počet translačných a rotačných smerov pohybu. Bežný kĺb má spravidla **3 stupne voľnosti**.

2. Kĺbová štruktúra a jej reprezentácia

Abstrakcia kĺbovej štruktúry sa v systémoch XR reprezentuje pomocou **acyklického grafu**, konkrétne ako **nepravidelný n-árny strom**.

- **Uzly stromu** reprezentujú samotné kĺby.
- **Spojnice (hrany)** reprezentujú kosti medzi kĺbmi.
- **Typy kĺbov:** Z hľadiska mechaniky ide o **translačné** (posuvné) a **rotačné** (otáčavé) kĺby. Pri modelovaní ruky sa rozlišujú špecifické typy ako MCP (kladkový), PIP (guľový) a DIP (sedlový).

3. Inverzná kinematika (IK)

Na rozdiel od priamej kinematiky, kde určujeme polohu z uhlov kĺbov, inverzná kinematika rieši opačnú úlohu: **hľadá konfiguráciu kĺbov (uhly) pre dosiahnutie požadovanej cieľovej polohy.**

- **Základná rovnica:** $f^{-1}(\text{poloha_koncového_efektora}) = \text{uhly_kĺbov}$.
- **Vlastnosti:** Proces je komutatívny z hľadiska výsledného bodu, ale nie z hľadiska trajektórie. Úloha nemusí byť vždy riešiteľná kvôli fyzikálnym limitom kĺbov alebo zložitosti inverzných matic.

4. Metódy riešenia IK

Zdroje uvádzajú dva základné prístupy k výpočtu:

- **Analytické metódy:** Poskytujú exaktné matematické vyjadrenie. Sú **veľmi rýchle a presné**, ale málo flexibilné, nakoľko sa nedajú jednoducho aplikovať na všeobecný (ľubovoľne dlhý) kinematický reťazec.
- **Numerické (iteratívne) metódy:** Sú pomalšie a menej presné, ale **vysoko adaptabilné** na rôzne štruktúry. Ich podstatou je postupné prechádzanie kĺbmi a korekcia ich polohy smerom k cieľu.

5. Algoritmus CCD (Cyclic Coordinate Descent)

Ide o populárnu numerickú metódu riešenia IK. Princíp spočíva v **opakovanom prechádzaní kĺbov v smere od koncového efektora k báze.**

1. Algoritmus začne pri koncovom efektore a prejde na nasledujúci kĺb v reťazci.
2. Aktuálny kĺb zmení svoj uhol tak, aby efektor smeroval bližšie k cieľu.
3. Tento proces sa opakuje, kým sa nedosiahne báza, čo predstavuje jednu **iteráciu**.

V praxi sa pre výpočty rotácií v IK systémoch často využívajú **quaternióny** v kombinácii s Eulerovým teorémom, čo zabraňuje strate stupňov voľnosti (gimbal lock).