

Univerzita Pardubice
Fakulta ekonomicko-správní

**Počítačová grafika a geometrické transformace
v rovině a prostoru**

Eva Hladíková

Bakalářská práce

2010

Univerzita Pardubice
Fakulta ekonomicko-správní
Ústav matematiky
Akademický rok: 2009/2010

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Eva HLADÍKOVÁ**
Studijní program: **B6209 Systémové inženýrství a informatika**
Studijní obor: **Informatika ve veřejné správě**

Název tématu: **Počítačová grafika a geometrické transformace v rovině a prostoru**

Z á s a d y p r o v y p r a c o v á n í :

Homogenní soustava souřadnic v rovině a prostoru
Transformace v rovině a prostoru, transformační matice
Projekce z prostoru do roviny
Vytvoření praktické aplikace v programu Maple pro práci s rovinnými transformacemi

Rozsah grafických prací:

Rozsah pracovní zprávy:

cca 30 stran

Forma zpracování bakalářské práce:

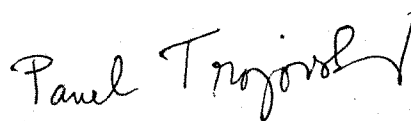
tištěná/elektronická

Seznam odborné literatury:

DE BERG, Mark, et al. Computational Geometry : Algorithms and Applications. 3rd rev. edition. Berlin : Springer-Verlag, 2008. ISBN 978-3-540-77973-5.

MARTIŠEK, Dalibor. Matematické principy grafických systémů. 1. vyd. Brno : Littera, 2002. ISBN 80-85763-19-2.

ŽÁRA, Jiří, et al. Moderní počítačová grafika : kompletní průvodce metodami 2D a 3D grafiky. 2. přepracované, rozšířené vyd. Praha : Computer Press, 2004. ISBN 80-251-0454-0.



Vedoucí bakalářské práce:

RNDr. PaedDr. Pavel Trojovský, Ph.D.
Ústav matematiky

Datum zadání bakalářské práce:

20. května 2009

Termín odevzdání bakalářské práce:

30. dubna 2010



doc. Ing. Renáta Myšková, Ph.D.

děkanka

L.S.



doc. RNDr. Bohdan Linda, CSc.

vedoucí ústavu

V Pardubicích dne 15. července 2009

Čestné prohlášení

Prohlašuji, že jsem tuto práci jsem vypracovala samostatně. Veškeré literární prameny a informace, které jsem v práci využila, jsou uvedeny v seznamu použité literatury.

Byla jsem seznámena s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích 27. 4. 2010

Eva Hladíková

Anotace

HLADÍKOVÁ, EVA. Počítačová grafika a geometrické transformace v rovině a prostoru. Pardubice : Fakulta ekonomicko-správní, Univerzita Pardubice, 2010. 54 s. Bakalářská práce.

Bakalářská práce se věnuje základním operacím v procesu počítačového zobrazování, a to geometrickým transformacím. V první kapitole je zmíněn způsob ukládání grafické informace a popis zobrazovacího řetězce. Druhá kapitola podrobně popisuje jednotlivé geometrické transformace. Projekcím z prostoru do roviny je věnována třetí kapitola. Jako ukázka aplikace geometrických transformací je nastíněno jejich prolínání do kartografických zobrazení. Finálním výstupem této práce je vizualizace algoritmů geometrických transformací v programu Maple.

Klíčová slova: počítačová grafika, geometrické transformace, projekce, algoritmy, kartografická zobrazení

Annotation

HLADÍKOVÁ, EVA. Computer graphics and geometric transformations in 2-D and 3-D space. Pardubice : Faculty of Economics and Administration, University of Pardubice, 2010, 54 pp. Bachelor Degree Thesis.

The bachelor degree thesis deals with basic operations in the process of computer imaging and especially geometrical transformations. In the first chapter the way of storing of graphic information and description of graphics pipeline is mentioned. The second chapter deals with individual geometrical transformations. The third chapter deals with projections of 3-D space to 2-D. As an illustration of application of geometrical transformations, insculation with cartographic notations is outlined. The final result of this work is a visualization of algorithms of geometrical transformations in the programme Maple.

Keywords: computer graphics, geometric transformations, projections, algorithms, cartographic projections

Poděkování

Mé díky patří RNDr. PaedDr. Pavlu Trojovskému, Ph.D. za čas věnovaný mé práci a také za velmi cenné rady, které mi byly vydatnou pomocí při zpracovávání této práce.

Obsah

Úvod	9
1 Model grafické informace v počítačové grafice	10
1.1 Grafická data	10
1.2 Digitální geometrie	11
1.3 Zobrazovací řetězec	12
1.3.1 Souřadné systémy v zobrazovacím řetězci	13
2 Geometrické transformace	14
2.1 Homogenní soustava souřadnic a transformační matice	15
2.2 Skládání transformací	16
2.3 Geometrické transformace v rovině	17
2.3.1 Posunutí	17
2.3.2 Otočení	18
2.3.3 Souměrnost - středová a osová	19
2.3.4 Změna měřítka	20
2.3.5 Zkosení	21
2.4 Geometrické transformace v prostoru	21
2.4.1 Posunutí	21
2.4.2 Otočení	21
2.4.3 Souměrnosti	23
2.4.4 Změna měřítka	23
2.4.5 Zkosení	24
2.5 Transformace souřadného systému	24
2.6 Geometrické transformace diskrétního obrazu	25
3 Projekce z prostoru do roviny	28
3.1 Obecný postup konstrukce promítání	29
3.2 Rovnoběžné promítání	30
3.2.1 Axonometrie	31
3.2.2 Kosoúhlé promítání	33
3.3 Středové promítání	34
3.4 Jednotné promítání	35
4 Aplikace geometrických transformací v kartografických zobrazeních	37
4.1 Geografická poloha	37

4.2	Kartografická zobrazení	38
4.3	Metody geografických transformací	39
4.3.1	Metoda tří parametrů	39
4.3.2	Metoda sedmi parametrů	40
4.4	Změna mapové projekce	40
4.5	Převzorkování rastrové reprezentace mapy	41
4.6	Výpočetní geometrie v geografických informačních systémech	42
5	Implementace geometrických transformací	
	v systému Maple	43
5.1	Charakteristika systému Maple	43
5.2	JavaView	43
5.3	Procedury a funkce pro geometrické transformace	44
5.3.1	Geometrické transformace v rovině	45
5.3.2	Geometrické transformace v prostoru	46
5.3.3	Knihovny a funkce programu Maple	47
5.4	Animace v programu Maple	48
5.5	Grafický výstup procedur pomocí JavaView	48
	Seznam použitých informačních zdrojů	51
	Seznam obrázků a tabulek	53
	Seznam příloh	54
	Přílohy	55

Úvod

Počítačová geometrie je jednou z aplikací počítačových systémů, jež vychází ze základů a terminologie matematické geometrie. Význam této disciplíny vzrostl s vývojem počítačové grafiky, CAD/CAM systémů a s potřebou vizualizace výsledků experimentálních měření a jevů v přírodních vědách.

Geometrické transformace tvoří jednu z nejzákladnějších skupin algoritmů pro zobrazování vůbec. Aplikují se všude tam, kde je potřeba polohová manipulace s obrazem nebo trojrozměrným modelem objektu. Z hlediska speciální aplikace v oboru mého studia, informatiky ve veřejné správě, stojí v popředí zájmu uplatnění geometrických transformací v geografických informačních systémech.

Cílem této bakalářské práce je zavést matematický aparát pro práci s geometrickými transformacemi, tedy matice transformací a homogenní soustavu souřadnic, popsat algoritmy geometrických transformací v rovině a prostoru včetně projekcí. Za specifickou aplikaci jsem zvolila ukázkou použití geometrických transformací v kartografii a tedy i v geografických informačních systémech. Protože se tyto algoritmy týkají zobrazování, dalším výstupem je vizualizace algoritmů pomocí CAS programu Maple a jejich export do webové stránky.

První kapitolou vstupuji do úplných základů modelu počítačové informace, kde vymezuji pojem pixelu, zpracování obrazu v zobrazovacím řetězci a souřadné systémy používané v počítačové grafice. Tato pasáž vymezuje roli operací geometrických transformací v zobrazovacím systému. Druhá kapitola již definuje pojem geometrická transformace a zavádí transformační matice a homogenní souřadnice. Hlavní obsah kapitoly tvoří popis a odvození jednotlivých druhů transformací v rovině a prostoru. Kapitolu uzavírá nástin problémů, které mohou na výstupu vznikat po provedení transformace, a některé metody jejich řešení. Třetí kapitola navazuje na předchozí zavedením projekčních metod pro zobrazení trojrozměrného modelu v rovině. Ve čtvrté kapitole se stručně věnuji souvislostem geometrických transformací a kartografických zobrazení. Poslední pátá kapitola popisuje implementaci algoritmů geometrických transformací pro jejich vizualizaci v programu Maple.

1 Model grafické informace v počítačové grafice

Obraz v počítačové grafice vychází z matematického modelu, jímž je ve dvourozměrném prostoru spojitá funkce $f(i, j)$ dvou souřadnic. [Hlav92] Tato kapitola shrnuje východiska tohoto modelu, jeho následné použití a transformaci pro získání fyzického výstupu obrazu. Pro zpracování grafických informací je důležité vědět, v jaké podobě jsou ukládány, proto se dále věnuji pojmu obrazového bodu a jeho rozlišení vzhledem k modelu a fyzickému výstupu. Následuje stručná charakteristika vybrané disciplíny počítačové grafiky, a to digitální geometrie, která velmi úzce souvisí s tématem mé bakalářské práce. Podkapitola Zobrazovací řetězec popisuje celkový proces zpracování obrazu ve svých významných fázích.

1.1 Grafická data

Nositeli grafické informace jsou grafická data. V závislosti na tom, jaké informace jsou v souborech ukládány, se obvykle dělí na data vektorová a rastrová (bitmapová).

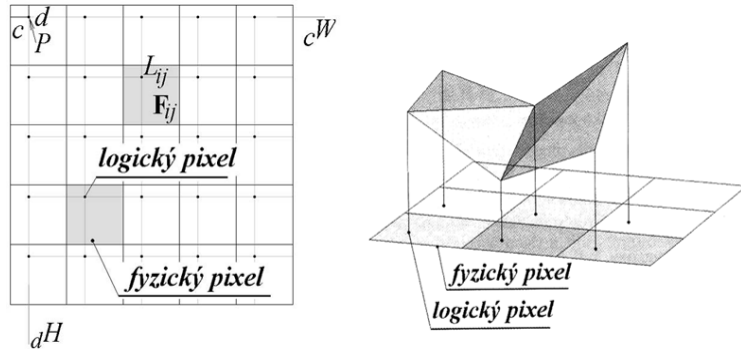
Vektorový grafický soubor obsahuje informace o objektu složeném z křivek a jednoduchých těles, které umožňují jeho konstrukci. Rovněž jsou o objektu například ukládány informace o barvě a tloušťce čáry. Základními geometrickými prvky (*graphics primitives*) jsou bod, linie a polygon. Na rozdíl od rastrových dat tedy neobsahuje informace o každém bodu tvořícím daný objekt. Tvoří ho seznam příkazů, podle nichž je žádaný obraz sestaven. [Mart02]

Rastrový grafický soubor je uložen jako matice, jejíž každý prvek odpovídá jednomu bodu obrazu. Právě matice jsou nejobvyklejší datovou strukturou pro reprezentaci obrazu nízké úrovně abstrakce. Rastrová data mohou dále obsahovat například informaci o způsobu komprese, velikosti obrazu nebo kódování barev. [Mart02, Buss03, Hlavá92]

Rastrová data jsou ukládána dle tradiční euklidovské geometrie do matice jako souřadnice bodů - objektů „bez rozměru“. Výstupní zařízení jsou však fyzické povahy a takové body nejsou schopny zobrazit. V počítačové grafice se proto vžil namísto pojmu bod pojem **pixel** jako nejmenší zobrazitelný útvar. S tímto pojmem se pracuje v rámci matematického modelování, kde je třeba rozlišovat mezi **fyzickým a logickým pixelem**. Ve smyslu fyzickém jsou data tvořena „množinou elementárních plošek“. Fyzický pixel je matematickým modelem pixelu, který abstrahuje od jednotlivých vlastností zobrazovacích zařízení a ponechává si jedinou, a to nenulové rozměry. Logické pixely jsou chápány jako množina izolovaných euklidovských bodů. Na tyto pixely se odkazujeme souřadnicemi¹. Běžně

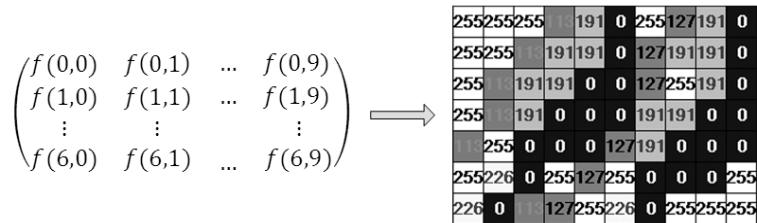
¹Je důležité, na který euklidovský bod fyzického pixelu odkazujeme. Tato specifikace se nazývá mapování. Vzájemné přiřazení mezi fyzickými a logickými pixely musí odpovídat skutečné velikosti a uspořádání fyzických pixelů.

je spíše používán pixel ve smyslu fyzickém. Vztah fyzického a logického pixelu ilustruje obrázek 1. Pixel reprezentuje 2-D grafiku a jeho paralelou v trojrozměrném prostoru je **voxel** jako nejmenší jednotka objemu. [Mart02]



Obrázek 1: Fyzický a logický pixel, zdroj [Mart02]

Pomocí souřadnic k pixelu pouze přistupujeme. To, co bodu teprve přiřazuje jeho vlastnosti jako je barva nebo propustnost světla, je **value**. Zjednodušeně je valuaace definovaná jako funkce z dvourozměrného pole nezáporných celých čísel do množiny, jež může být značně obecně zadána. Může obsahovat čísla typu *float* nebo vektory. Například barevný obrázek v RGB je reprezentován pixely s hodnotami vektoru o třech složkách (*red*, *green*, *blue*). Obrázek 2 uvádí příklad, kde oborem hodnot valuaace je 256 prvková množina všech barev grafického formátu GIF. [Mart02, Klet04]



Obrázek 2: Valuaace formátu GIF, zdroj vlastní, [Mart02]

1.2 Digitální geometrie

Digitální geometrie se zabývá studiem geometrických a topologických vlastností souboru pixelů nebo voxelů. Jejím cílem je získat kvantitativní informace o objektech analýzou digitalizovaných obrazů, které vznikly nahrazením objektu diskretní množinou bodů uspořádaných v mřížce. [Klet04]

Předpokládáme, že objekty jsou reprezentovány spojitou množinou pixelů a dále to, že sledované vlastnosti zahrnují vlastnosti studované v euklidovské geometrii nebo geometrii

podobností. Potom můžeme přijmout výpočty digitální geometrie jako adekvátní aproximace závěrů geometrie podobností. V tomto smyslu pohlížíme na digitální geometrii jako digitalizovanou geometrii podobností. [Klet04]

Následuje výčet příbuzných matematických disciplín a témat digitální geometrie, jež jsou stěžejní pro téma mé práce.

- Metrické prostory a souřadné systémy
- Euklidovská a afinní geometrie, geometrie podobností
- Lineární algebra
- Projektivní geometrie

Procesy projektivní geometrie (promítání) mapují viditelnou část scény v prostoru na rovinu. Při promítání se ztrácí prostorové informace, a proto je zavedena celá řada postupů a způsobů promítání pro zvýšení reálného vjemu promítaného objektu a odvození jeho různých prostorových vztahů. Podrobně se této problematice věnuji ve třetí kapitole.

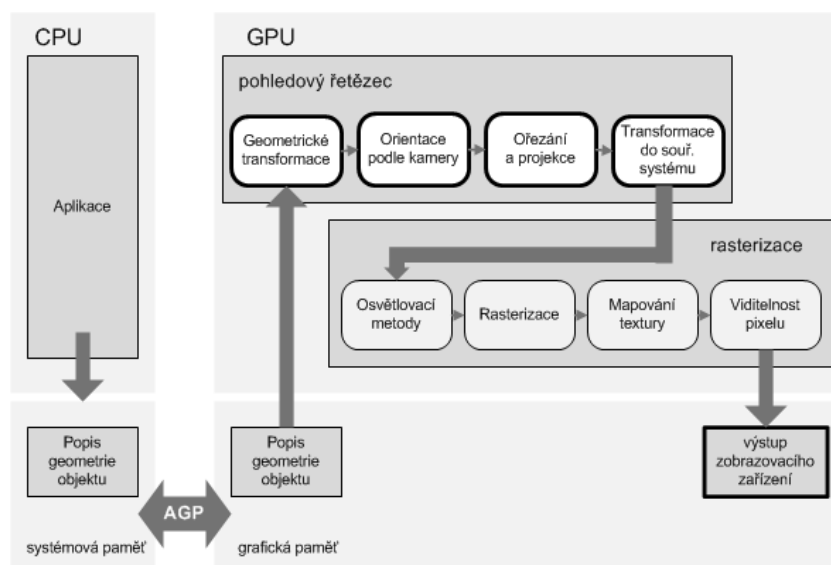
Digitální geometrie se dále opírá o poznatky kombinatorické a výpočetní geometrie, topologie, teorie grafů, fuzzy geometrie, integrální geometrie, stereologie a tomografie. [Žára04, Klet04]

1.3 Zobrazovací řetězec

Postupné provádění dílčích operací při zobrazování se nazývá zobrazovací řetězec (*rendering pipeline*), jehož podmnožinou je pohledový řetězec (*viewing pipeline*). Pohledový řetězec zajišťuje zejména promítání a popis metod v tomto řetězci je jedním z cílů této bakalářské práce. [Žára04]

Vstupní proud dat popisujících geometrii objektů (typicky jde o proud trojúhelníků) je podroben pohledové transformaci, poté jsou z dalšího zpracování vyloučeny ty části objektu, které leží mimo pohledový objem. Následující transformace převede data do souřadnicového systému obrazovky. Lokální osvětlení řeší barevné stínování na jednotlivých ploškách, na ně jsou následně nanесeny textury a poté jsou zaznamenány jako pixely, neboli jsou rastrovány (s respektováním viditelnosti). [Žára04]

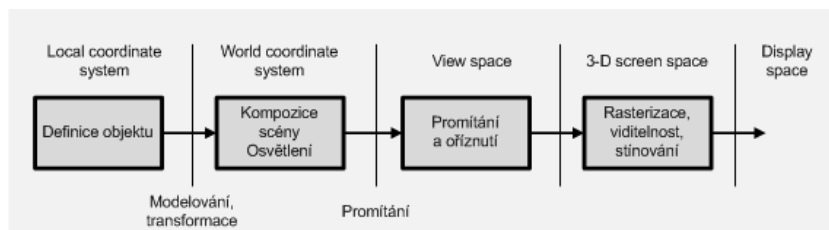
Scéna zachycující jednu z možných variant zobrazovacího řetězce je znázorněna na obrázku 3. V pohledovém řetězci jsou geometrické transformace sice uvedeny na jeho začátku, ale metody těchto transformací procházejí v průběhu celého pohledového řetězce. Realizace zobrazovacího řetězce v oblasti rasterizace nemusí vždy odpovídat tomuto schématu, záleží na povaze úlohy zobrazení a její rozsáhlosti. [Žára04]



Obrázek 3: Zobrazovací řetězec, zdroj vlastní, [Nvid09]

1.3.1 Souřadné systémy v zobrazovacím řetězci

Grafická informace je v procesu zobrazovacího řetězce transformována z jednoho prostoru do jiného. **Local coordinate system** (modelové souřadnice) je prostor pro modelování objektu. Jakmile je objekt vymodelován, umístíme jej do scény, kterou chceme převést na reálný obraz (*rendering*). Všechny objekty tvořící tuto scénu jsou umístěny ve svých vlastních lokálních systémech a systém, který je globální pro všechny, se nazývá **world coordinate system** (světové souřadnice). Do něj jsou všechny objekty transformovány. Pro potřebu promítání se zavádí **view space** (pohledové souřadnice), ve kterém musí být definovány parametry promítání, jako je například střed a směr promítání. Nakonec se promítnutá scéna ořeže a skryjí se zakryté body v **3-D screen space** a odtud se scéna transformuje do finálního dvourozměrného systému souřadnic pro zobrazení na výstupu. [Watt00]



Obrázek 4: Souřadné systémy, zdroj vlastní, [Watt00]

2 Geometrické transformace

V této kapitole se v úvodu podrobně věnuji zavedení geometrických transformací pomocí matic a homogenních souřadnic a následně popisu jednotlivých transformací.

Geometrické transformace spolu s projekcemi jsou všudypřítomnými součástmi počítačové grafiky a CAD systémů. Hlavní účel těchto transformací spočívá v poskytování metod pro změny tvarů a polohy objektů. Jejich použití směřuje k zjednodušení řešení problémů geometrického modelování. Jestliže například obrázek obsahuje dvě identické (nebo i jen podobné) části, pak stačí, když bude sestrojena pouze jedna z nich. Druhou získáme zkopírováním první, následně jejím posunutím a otočením, čímž vykreslíme část obrázku v tom správném tvaru, velikosti, pozici a orientaci. Základní transformace jsou **otáčení, posunutí, změna měřítka, zkosení a operace vzniklé jejich skládáním**. [Salo06, Žára04, Buss03]

Geometrická transformace je funkce, jejíž definiční obor a obor hodnot je množina bodů. V rovině kartézské souřadné soustavě mějme body $P[p_1; p_2]$, $P'[p'_1; p'_2]$ a zobrazení $\mathcal{Z} : P \rightarrow P'$. Toto zobrazení lze zadat soustavou rovnic

$$p'_1 = f(p_1; p_2) \quad p'_2 = g(p_1; p_2)$$

Transformaci $\mathcal{Z}$ nazveme **lineární**, jestliže výše uvedená soustava je lineární, tj. jestliže vztah mezi souřadnicemi bodů P, P' lze vyjádřit soustavou lineárních rovnic

$$\begin{aligned} p'_1 &= z_{11}p_1 + z_{12}p_2 \\ p'_2 &= z_{21}p_1 + z_{22}p_2 \end{aligned} \Rightarrow \begin{pmatrix} p'_1 \\ p'_2 \end{pmatrix} = \begin{pmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \Leftrightarrow \mathbf{P}' = \mathbf{Z} \cdot \mathbf{P}.$$

Pokud výše uvedenou soustavu rozšíříme přičtením vektoru posunutí $\vec{t} = (t_x; t_y)$, získáme transformaci **afinní**. Jinými slovy je afinní transformace složením lineární transformace a translace.

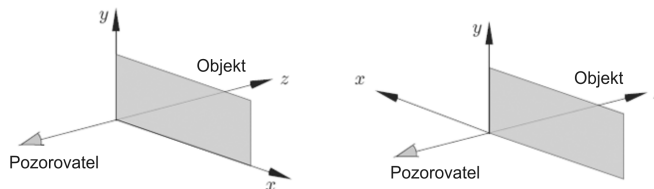
$$\begin{aligned} p'_1 &= z_{11}p_1 + z_{12}p_2 + t_x \\ p'_2 &= z_{21}p_1 + z_{22}p_2 + t_y \end{aligned} \Rightarrow \begin{pmatrix} p'_1 \\ p'_2 \end{pmatrix} = \begin{pmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \Leftrightarrow \mathbf{P}' = \mathbf{Z} \cdot \mathbf{P} + \vec{t}. \quad (1)$$

Transformace $\mathcal{Z}$ je tak určena maticí $\mathbf{Z}$ a vektorem $\vec{t}$. Tuto matici $\mathbf{Z}$ nazýváme **transformační maticí**. Ka každé transformační matici, která je regulární, existuje matice k ní inverzní, což má svůj významný účel při skládání transformací. [Mart02, Klet04]

Transformace může být aplikována na každý pixel objektu. Mnohdy je výhodnější použít transformaci pouze na klíčové body objektu, které jej definují. Obraz je pak zkonstruován z transformovaných klíčových bodů. Pro transformace v prostoru platí stejné principy. [Salo06]

2.1 Homogenní soustava souřadnic a transformační matice

Před vymezením významu pojmu homogenní soustavy souřadnic je třeba se vrátit ke specifiku kartézského souřadného systému tří rozměrů. Ve dvou dimenzích existuje pouze jeden kartézský systém o dvou kolmých osách x a y . Souřadný systém tří rozměrů podobně tvoří tři kolmé osy x , y a z . Existují však dva rozdílné typy tohoto systému, pravotočivý nebo levotočivý kartézský systém souřadnic, jak ukazuje obrázek 5. Levotočivý systém souřadnic je možné definovat jako zrcadlový obraz pravotočivého systému. Nelze však jeden systém transformovat v druhý pouhou translací nebo rotací. Rozdíl mezi těmito systémy se stává významný ve chvíli, kdy je 3-D objekt promítán na 2-D obrazovku. Objekt je definován ve světovém systému souřadnic a ten je podle konvence pravotočivý. Levotočivý se používá ve speciálním kontextu počítačového promítání. Transformace bodů do soustavy homogenních souřadnic zaručí správnou reprezentaci obrazu nezávisle na typu systému. [Salo06, Hear97, Watt00]



Obrázek 5: Levotočivý a pravotočivý systém souřadnic, zdroj [Salo06]

Výhodou použití homogenních souřadnic je zjednodušení výpočtů. Homogenní souřadnice na rozdíl od nehomogenních umožňují vyjádření nejčastěji používaných transformací pomocí jediné matice. Další významnou transformací vyjádřenou v těchto souřadnicích je perspektivní promítání, jež bude popsáno v následující kapitole. Tento způsob vyjádření dovoluje efektivně pracovat s maticovými operacemi, a tak moderní grafické procesory dokáží rychleji zpracovat scénu. [Žára04]

Definice (homogenní souřadnice): Uspořádanou trojici $(\xi_1; \xi_2; \xi_3)$, $\xi_3 \neq 0$, nazýváme homogenní souřadnice bodu M v rovině, jestliže platí $\frac{\xi_1}{\xi_3} = x$, $\frac{\xi_2}{\xi_3} = y$, kde $[x, y]$ jsou kartézské souřadnice bodu M v rovině.²

Soustava rovnic

$$\begin{pmatrix} \xi'_1 \\ \xi'_2 \\ \xi'_3 \end{pmatrix} = \begin{pmatrix} z_{11} & z_{12} & z_{13} \\ z_{21} & z_{22} & z_{23} \\ z_{31} & z_{32} & z_{33} \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}$$

²Číslo $\xi_3 \neq 0$ může být zvoleno libovolné reálné, ale pro jednoduchost jej nejčastěji volíme rovno jedné. V homogenních souřadnicích má tedy bod nekonečně mnoho reprezentací, které dohromady v prostoru vyplní přímku.

přejde v soustavu (1), jestliže $z_{31} = z_{32} = 0, z_{33} = \xi_3 = 1$.

$$\begin{aligned} p'_1 &= z_{11}p_1 + z_{12}p_2 + t_x \\ p'_2 &= z_{21}p_1 + z_{22}p_2 + t_y \\ 1 &= 0p_1 + 0p_2 + 1 \end{aligned} \Rightarrow \begin{pmatrix} p'_1 \\ p'_2 \\ 1 \end{pmatrix} = \begin{pmatrix} z_{11} & z_{12} & t_x \\ z_{21} & z_{22} & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ 1 \end{pmatrix}$$

$$\Rightarrow \mathbf{P}' = \mathbf{Z}_{3 \times 3} \cdot \mathbf{P}.$$

Transformace $\mathcal{Z}$ je pak určena jedinou maticí $\mathbf{Z}$ typu 3×3 . Pro obecnou matici reprezentující transformaci bodu $P = [p_1; p_2; p_3; 1]$ třírozměrného prostoru v homogenních souřadnicích na bod $P' = [p'_1; p'_2; p'_3; 1]$ platí

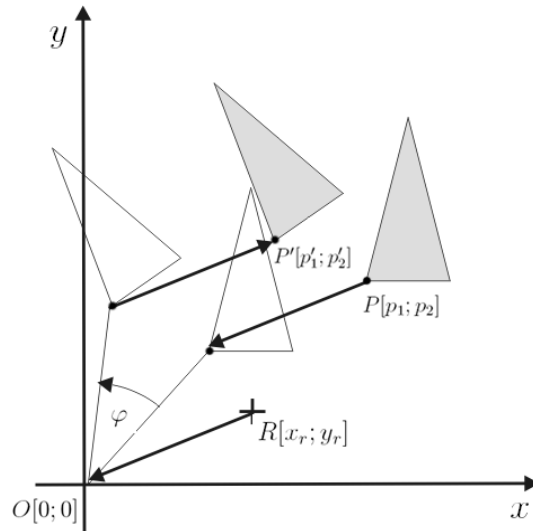
$$\begin{pmatrix} p'_1 \\ p'_2 \\ p'_3 \\ 1 \end{pmatrix} = \begin{pmatrix} z_{11} & z_{12} & z_{13} & z_{14} \\ z_{21} & z_{22} & z_{23} & z_{24} \\ z_{31} & z_{32} & z_{33} & z_{34} \\ z_{41} & z_{42} & z_{43} & z_{44} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ 1 \end{pmatrix} \Rightarrow \mathbf{P}' = \mathbf{Z}_{4 \times 4} \cdot \mathbf{P}.$$

Poslední řádek matice $\mathbf{Z}$ je určen pro použití projekce a můžeme jej také vyjádřit jako $(P_x, P_y, P_z, 1)$. Poslední sloupec matice, neboli $(t_x, t_y, t_z, 1)^T$, definuje posunutí ve směru vektoru $\vec{t}$. [Mart02, Salo06]

2.2 Skládání transformací

Transformaci vzniklou složením z více transformací lze vyjádřit jedinou maticí, již získáme postupným násobením matic reprezentujících dílčí transformace. Násobení matic ale není komutativní operace, proto záleží na pořadí transformací, které chceme provést. Například rotace kolem počátku následovaná translací není totéž jako translace následovaná rotací. Často používané složené transformace je výhodné předem sestavit, neboť opakované násobení matic by bylo při zpracování většího objemu dat neefektivní. Grafické knihovny, jako např. OpenGL, tuto možnost poskytují.

Jak již bylo dříve uvedeno, ke každé regulární transformační matici existuje matice k ní inverzní. Složením matice $\mathbf{Z}$ a matice $\mathbf{Z}^{-1}$ získáme identitu. Toho lze využít při takových úlohách, jako je otočení kolem obecného bodu $R[x_r; y_r]$ o úhel φ , při němž použijeme složenou transformaci $P' = \mathbf{T}(x_r; y_r) \cdot \mathbf{R}(\varphi) \cdot \mathbf{T}(-x_r; -y_r) \cdot P$. Objekt posuneme tak, aby střed otáčení byl v počátku, otočíme jej o úhel φ a posuneme otočený objekt zpět. Tento postup ilustruje obrázek 6. Zde je třeba upozornit na to, že pozdější transformace se v tomto symbolickém zápisu uvádějí zleva. [Salo06, Žára04]

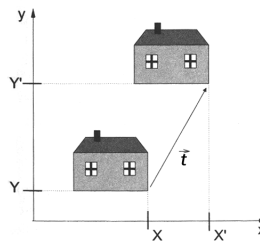


Obrázek 6: Rotace kolem obecného bodu - skládání transformací, zdroj vlastní

2.3 Geometrické transformace v rovině

2.3.1 Posunutí

Aplikací posunutí (translace) docílíme změny pozice objektu v souřadném systému. Jak znázorňuje obrázek 7, bod $P[p_1; p_2]$ zobrazíme v transformaci $\mathcal{T}$ na bod $P'[p'_1; p'_2]$ přičtením vektoru posunutí $\vec{t} = (t_x; t_y)$.



Obrázek 7: Posunutí ve 2-D, zdroj [Žára04]

$$p'_1 = p_1 + t_x, \quad p'_2 = p_2 + t_y$$

Po převodu do homogenních souřadnic získáváme transformaci posunutí $\mathcal{T}$ s transformační maticí $\mathbf{T}(t_x; t_y)$, resp. s maticí inverzní $\mathbf{T}^{-1}(t_x; t_y)$

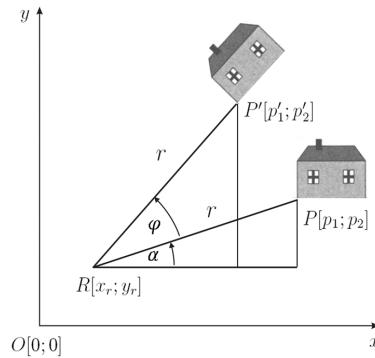
$$P' = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \cdot P = \mathbf{T}(t_x; t_y) \cdot P, \quad \mathbf{T}^{-1}(t_x; t_y) = \mathbf{T}(-t_x; -t_y) = \begin{pmatrix} 1 & 0 & -t_x \\ 0 & 1 & -t_y \\ 0 & 0 & 1 \end{pmatrix}.$$

Transformace translace mění polohu objektu, aniž by docházelo k jeho deformaci. Každý bod je posunut o stejnou vzdálenost. Aplikací posunutí na úsečku dojde k transformaci

pouze jejich koncových bodů a dle nich se zbylé nově zkonstruují. Polygon je posunut přičtením vektoru posunutí ke každému z jeho vrcholů. Podobně transformujeme i křivky pomocí jejich definičních bodů. Stejným způsobem pracují s těmito základními geometrickými útvary - grafickými primitivy a jejich významnými body i ostatní geometrické transformace, aplikuje se na ně samozřejmě už jiná transformační matice. [Hear97, Žára04]

2.3.2 Otočení

Dvourozměrná rotace je aplikována na objekt jeho změnou pozice po kružnici. K definování rotace specifikujeme úhel otočení φ a pozici středu otočení $R[x_r; y_r]$, kolem kterého je objekt otáčen. Kladná hodnota úhlu otočení určuje rotaci proti směru hodinových ručiček, naopak záporná po směru. Transformaci $\mathcal{R}$ ilustruje obrázek 8.



Obrázek 8: Otáčení ve 2-D, zdroj vlastní

Užitím trigonometrie jsou transformované body vyjádřeny jako

$$\begin{aligned}x' &= r \cos(\alpha + \varphi) = r \cos \alpha \cos \varphi - r \sin \alpha \sin \varphi \\y' &= r \sin(\alpha + \varphi) = r \cos \alpha \sin \varphi + r \sin \alpha \cos \varphi\end{aligned}$$

K odvození rovnice transformace rotace $\mathcal{R}$ jsou použity původní souřadnice bodu v polárních souřadnicích

$$x = r \cos \alpha, \quad y = r \sin \alpha,$$

kde r je konstantní vzdálenost bodu od počátku a α je původní odchylka bodu od horizontální osy. Ze substituce těchto souřadnic do rovnic pro transformované body plyne

$$\begin{aligned}x' &= x \cos \varphi - y \sin \varphi \\y' &= x \sin \varphi + y \cos \varphi\end{aligned}$$

Odtud pro otočení objektu kolem počátku soustavy souřadnic $O[0;0]$ s transformační maticí otáčení $\mathbf{R}$, resp. inverzní maticí $\mathbf{R}^{-1}$ platí

$$\mathbf{R}(O, \varphi) = \begin{pmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{R}^{-1}(O, \varphi) = \mathbf{R}(O, -\varphi) = \begin{pmatrix} \cos \varphi & \sin \varphi & 0 \\ -\sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

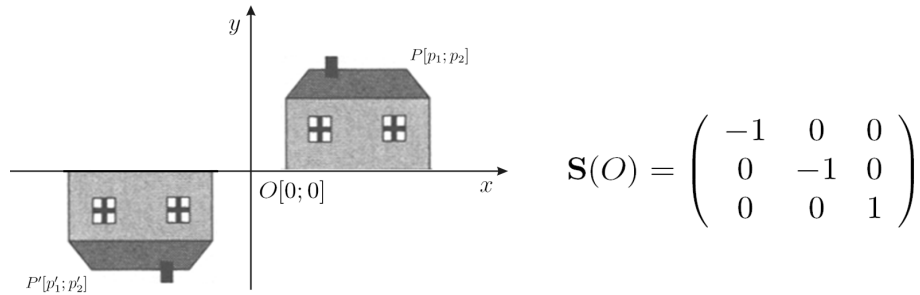
Otočení bodu v rovině kolem libovolného bodu $R[x_r; y_r]$ můžeme získat následující posloupností transformací. Posuneme objekt tak, že střed otočení R bude ležet v počátku soustavy. Otočíme objekt kolem počátku a objekt posuneme zpět, aby se bod R přesunul na svoji původní souřadnici. [Hear97, Žára04]

$$\mathbf{R}(R, \varphi) = \mathbf{T}(x_r; y_r) \cdot \mathbf{R}(\varphi) \cdot \mathbf{T}(-x_r; -y_r)$$

2.3.3 Souměrnost - středová a osová

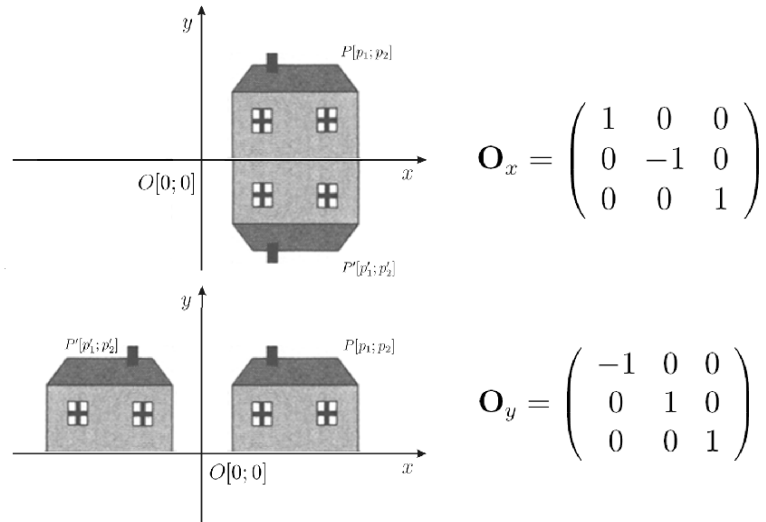
Obě souměrnosti jsou transformacemi, které produkují zrcadlový obraz objektu. Souměrnosti jsou také zvláštním případem změny měřítka, kde absolutní hodnota koeficientu měřítka je rovna jedné. Pro středovou souměrnost se středem v počátku platí $s_x = -1$ a $s_y = -1$. Souměrnost podle osy x má koeficienty $s_x = -1$ a $s_y = 1$, obdobně pro osu y je $s_x = 1$ a $s_y = -1$.

Středová souměrnost $\mathcal{S}$ je speciálním případem rotace o 180° okolo středu souměrnosti. Obraz bodu objektu má pak souřadnice s opačnými znaménky. Dále uvádím příklad této souměrnosti podle počátku souřadnic O a její transformační matici.



Obrázek 9: Středová souměrnost ve 2-D podle počátku a její matice transformace, zdroj vlastní

Základní podobou osové souměrnosti $\mathcal{O}$ je transformace podle souřadné osy x nebo y . Souměrnost v rovině xy podle osy y , resp. x zachovává druhou souřadnici, resp. první souřadnici bodu. Grafická znázornění těchto transformací s odpovídajícími transformačními maticemi ilustruje obrázek 10. Lze však sestavit obraz v souměrnosti podle jakékoli přímky (osy souměrnosti) $y = mx + n$ kombinací translace, rotace a osové souměrnosti aplikované na tuto přímku tak, abychom ji ztotožnili s jednou ze souřadných os, a jejím zpětným posunutím do původní polohy. [Hear97, Žára04]



Obrázek 10: Osové souměrnosti ve 2-D a jejich matice transformace, zdroj vlastní

2.3.4 Změna měřítka

Pomocí změny měřítka $\mathcal{H}$ (*scaling*) současně měníme polohu i velikost objektu ve směru souřadných os. Tato transformace je definována koeficienty s_x a s_y a obrazy bodu získáme vynásobením jejich původních souřadnic těmito koeficienty, tedy

$$p'_1 = p_1 \cdot s_x, \quad p'_2 = p_2 \cdot s_y.$$

Pokud je $|s_x| < 1$, dojde ke zmenšení a přiblížení ve směru osy x transformovaného objektu k počátku souřadnic (nebo k nějakému pevně zvolenému bodu). Je-li $|s_x| > 1$, dojde k prodloužení. Analogie platí i pro koeficient s_y . Znaménko koeficientu měřítka může být zvoleno i záporné, pak dochází k prodloužení či zmenšení v opačném směru. Jestliže $s_x = s_y$, mluvíme o rovnoměrném měřítku (*uniform scaling*) a proporce objektu jsou zachovány. Příslušné transformační a inverzní matice vzhledem k počátku souřadnic mají tvar

$$\mathbf{H}(O, s_x, s_y) = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{H}^{-1}(O, s_x, s_y) = \mathbf{H}(O, 1/s_x, 1/s_y) = \begin{pmatrix} 1/s_x & 0 & 0 \\ 0 & 1/s_y & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Místo počátku souřadnic můžeme zvolit libovolný bod $S[x_h; y_h]$, vzhledem k němuž se transformace provede. Podobně jako u rotace se skládají základní transformace v daném pořadí: posuneme objekt tak, že fixní bod S se ztotožní s počátkem soustavy. Provedeme změnu měřítka vzhledem k počátku soustavy a užitím inverzní translace prvního kroku přesuneme objekt zpátky do původní pozice. [Hear97, Žára04]

$$\mathbf{H}(S, s_x, s_y) = \mathbf{T}(x_h; y_h) \cdot \mathbf{H}(S, s_x, s_y) \cdot \mathbf{T}(-x_h; -y_h)$$

2.3.5 Zkosení

Transformace zkosení $\mathcal{A}$ (*shear*) zkresluje tvar objektu. Dvě běžná zkosení jsou taková, která mění pouze x -ové, nebo pouze y -ové souřadnice ve směru dané souřadné osy. Osa, která může být tedy souřadnou osou x nebo y (obecně ale může být jako osa opět zvolena libovolná referenční přímka), je určující. Dále pro zkosení definujeme parametry zkosení sh_x a sh_y .

Leží-li osa zkosení v souřadné ose x , resp. v souřadné ose y , pak

$$\mathbf{A}_x(sh_x) = \begin{pmatrix} 1 & sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \text{resp. } \mathbf{A}_y(sh_y) = \begin{pmatrix} 1 & 0 & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Pro matice inverzní platí $\mathbf{A}_x^{-1}(sh_x) = \mathbf{A}_x(-sh_x)$, $\mathbf{A}_y^{-1}(sh_y) = \mathbf{A}_y(-sh_y)$. [Hear97, Žára04]

2.4 Geometrické transformace v prostoru

Metody pro geometrické transformace 3-D prostoru jsou rozšířením metod dvourozměrných tím, že do nich zahrneme souřadnici z . Translace a změna měřítka jsou toho jednoduchým příkladem. Složitější operace však nastávají při rotaci. Stejně jako v případě dvourozměrných transformací se účelně používají homogenní souřadnice a jedna maticová forma složená v určitém pořadí z jednotlivých transformačních matic. [Salo06]

2.4.1 Posunutí

Posunutí v prostoru je určeno vektorem posunutí $\vec{t} = (t_x; t_y; t_z)$. Z následujících rovnic plyne rovnice translace. [Hear97, Žára04]

$$p'_1 = p_1 + t_x, \quad p'_2 = p_2 + t_y, \quad p'_3 = p_3 + t_z$$

$$P' = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot P = \mathbf{T}(t_x; t_y; t_z) \cdot P, \quad \mathbf{T}^{-1}(t_x; t_y; t_z) = \mathbf{T}(-t_x; -t_y; -t_z)$$

2.4.2 Otočení

Otočení kolem jednotlivých souřadnicových os může být jedním z případů otočení ve třech rozměrech. Matice $\mathbf{R}_x$ reprezentující otáčení okolo osy x o úhel α a odpovídající inverzní matice jsou

$$\mathbf{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{R}_x^{-1}(\alpha) = \mathbf{R}_x(-\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Dále jsou sestaveny matice pro otočení kolem osy y o úhel β a kolem osy z o úhel γ . [Žára04]

$$\mathbf{R}_y(\beta) = \begin{pmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{R}_z(\gamma) = \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 & 0 \\ \sin \gamma & \cos \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Otočení kolem osy z je pouhým rozšířením rotace v rovině ve trojrozměrném prostoru.

$$\begin{aligned} x' &= x \cos \gamma - y \sin \gamma \\ y' &= x \sin \gamma + y \cos \gamma \\ z' &= z \end{aligned}$$

Transformační rovnice pro rotace okolo zbylých os vycházejí z cyklické permutace souřadnicových parametrů x, y, z . To znamená přechod $x \rightarrow y \rightarrow z \rightarrow x$.

Pro rotaci okolo osy x substitucí dostáváme rovnice

$$\begin{aligned} y' &= y \cos \alpha - z \sin \alpha \\ z' &= y \sin \alpha + z \cos \alpha \\ x' &= x \end{aligned}$$

Transformační rovnice pro otočení okolo osy y jsou vyjádřeny analogicky.

Rotační matice pro kteroukoli osu, která neleží v nějaké souřadné ose, může být složena z translace a rotace kolem os. Žádanou matici získáme sestavením posloupnosti transformací, která nastaví osu rotace do polohy splývající s jednou souřadnou osou. Pak provedeme transformaci kolem souřadné osy o daný úhel rotace. Posledním krokem je provedení inverzní posloupnosti transformací, jež vrátí osu rotace do původní polohy. Ve speciálním případě, kde je objekt otočen okolo osy rovnoběžné s některou ze souřadných os, rovnice rotace zní $\mathbf{R}(\varphi) = \mathbf{T}^{-1} \cdot \mathbf{R}_\lambda(\varphi) \cdot \mathbf{T}$. Pokud uvažujeme o obecném případě umístění osy rotace, maticová forma $\mathbf{R}(\varphi)$ je rovna výsledku složení následujících transformací. Objekt posuneme tak, aby osa rotace procházela počátkem souřadnic, pak jej otočíme, aby osa rotace splynula s jednou souřadnou osou (nejčastěji z). Provedeme danou rotaci okolo osy a aplikujeme inverzní rotaci a translaci objektu pro navrácení osy do původní polohy. Transformační matice pro rotaci kolem libovolné osy je vyjádřena jako kompozice sedmi transformací, tedy $\mathbf{R}(\varphi) = \mathbf{T}^{-1} \cdot \mathbf{R}_x^{-1}(\alpha) \cdot \mathbf{R}_y^{-1}(\beta) \cdot \mathbf{R}_z(\gamma) \cdot \mathbf{R}_y(\beta) \cdot \mathbf{R}_x(\alpha) \cdot \mathbf{T}$. [Hear97]

Rotační matice určené osami souřadnic jsou jednoduché, ale obecně nepříliš užitečné, protože v praxi zřídka víme, jak vhodně určit jednotlivé matice, resp. konkrétní dílčí Eulerovy úhly α , β a γ . Jsou však oblasti jako astronautika a letectví³, ve kterých je otočení okolo souřadných os běžné. Tímto způsobem s rotací pracují i L-systémy objevené za účelem popisu morfologie rostlin. Nakonec rotaci kolem os využívá projekce axonometrie. [Salo06]

K reprezentaci rotací ale nejsou matice vždy nejvýhodnější. Rotační úlohu však lze převést na promítání a skládání několika vektorů. Následně místo s maticemi pracujeme s **kvaterniony** - analogií komplexních čísel v prostoru, a tedy i s vektory. Existuje vztah pro převod kvaternionu na rotační matici, protože mnoho grafických knihoven a hardware používají rotace v maticovém tvaru. Tato problematika obsahuje více matematických podrobností, proto pouze odkáží na odbornou publikaci [Watt00]. Rotace pomocí matic obsahuje nadbytečné informace a násobení rotačních matic vyžaduje více aritmetických operací než násobení kvaternionů. Další význam kvaternionů tkví v použití v animacích a sférické lineární interpolaci mezi dvěma 3-D rotacemi. [Žára04] Rozbor úlohy rotace v prostoru a její realizaci v programu Maple je uveden v příloze A.

2.4.3 Souměrnosti

Na souměrnosti ve 3-D můžeme nahlížet stejně jako ve 2-D jako na speciální případ změny měřítka s koeficienty $|s_x| = |s_y| = |s_z| = 1$. Na rozdíl od dvourozměrného případu můžeme kromě osové a středové souměrnosti uvažovat o případě souměrnosti podle roviny. Nejčastěji se provádí souměrnost podle rovin xy , xz nebo yz , jež není ničím jiným, než pouhou změnou znaménka jedné ze souřadnic bodu. Speciálně lze souměrnost podle roviny xz považovat za konverzi mezi pravotočivou a levotočivou soustavou souřadnic. Souměrnost můžeme také provést podle libovolné roviny jako kombinaci rotací a souměrností podle zmíněných souřadných rovin. [Hear97, Žára04, Mart02, Salo06]

2.4.4 Změna měřítka

Transformace změny měřítka je určena měřítkovými koeficienty $s_x \neq 0$, $s_y \neq 0$ a $s_z \neq 0$, jejichž hodnoty způsobují změnu objektu ve směru příslušné souřadnicové osy. Rovnice

$$p'_1 = p_1 \cdot s_x, \quad p'_2 = p_2 \cdot s_y, \quad p'_3 = p_3 \cdot s_z$$

udávají změnu rozměru a polohy objektu vzhledem k počátku souřadnic. Jestliže si jsou koeficienty měřítka rovny, tvar objektu zůstává zachován. [Hear97, Žára04]

³Eulerovy úhly slouží k určení relativní orientace letadla a odpovídají jim odborné termíny *yaw*, *pitch* a *roll*.

$$\mathbf{O}_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{O}_y = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{O}_z = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{O}_{xy} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{O}_{yz} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{O}_{xz} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{S}(O) = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{H}(O, s_x, s_y, s_z) = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{H}^{-1}(O, s_x, s_y, s_z) = \mathbf{H}(O, 1/s_x, 1/s_y, 1/s_z)$$

2.4.5 Zkosení

Transformace zkosení modifikuje tvar objektu a může být využita také v promítání pro získání základních projekčních transformací. Operaci zkosení ve třech rozměrech můžeme rozdělit na tři případy ve směru jednotlivých rovin xy, xz a yz . Ve všech třech případech určují koeficienty sh_x, sh_y a sh_z míru zkosení v odpovídajícím směru. Kromě těchto základních možností lze provést zkosení jejich kombinací. Tyto pokročilé operace jsou ovládány až šesti parametry. [Hear97, Žára04, Salo06]

$$\mathbf{A}_{yz} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ sh_y & 1 & 0 & 0 \\ sh_z & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{A}_{xz} = \begin{pmatrix} 1 & sh_x & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & sh_z & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{A}_{xy} = \begin{pmatrix} 1 & 0 & sh_x & 0 \\ 0 & 1 & sh_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

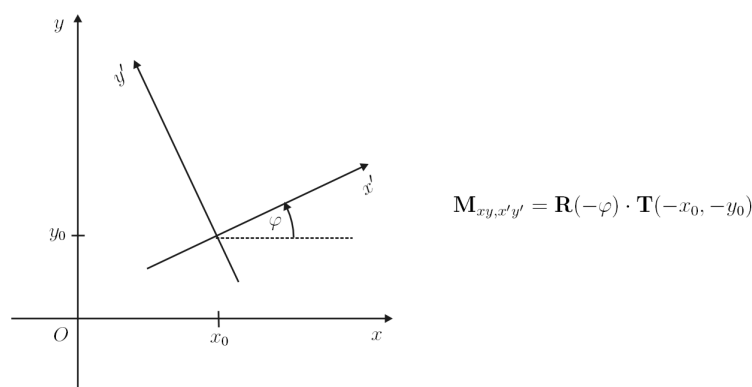
2.5 Transformace souřadného systému

Dosud jsem popisovala metody pro transformaci množiny bodů objektu na jinou množinu bodů, kde obě sady pocházely ze stejného souřadného systému. Dle tohoto předpokladu souřadný systém zůstal nezměněn a objekt byl transformován například ve vztahu k počátku souřadnic. Alternativní, ale ekvivalentní možnost jak přemýšlet o transformacích, je změna samotného souřadného systému. Tento náhled na problém je užitečný v případě,

kdy několik objektů, každý definovaný ve svém lokálním souřadném systému, má být zkombinováno a my si přejeme vyjádřit souřadnice těchto objektů v jednom globálním souřadném systému. [Fole94]

Původní lokální souřadný systém nemusí být ani kartézský, ale cílový globální systém pro finální zobrazení již kartézský být musí. Příkladem soustav souřadnic, které nejsou kartézskými, je polární, eliptická nebo parabolická soustava. [Hear97]

Transformaci jednoho kartézského souřadného systému do druhého v 2-D provedeme translací počátku souřadnic jednoho systému do počátku druhého a otočením souřadných os prvního tak, aby s cílovými splynuly.



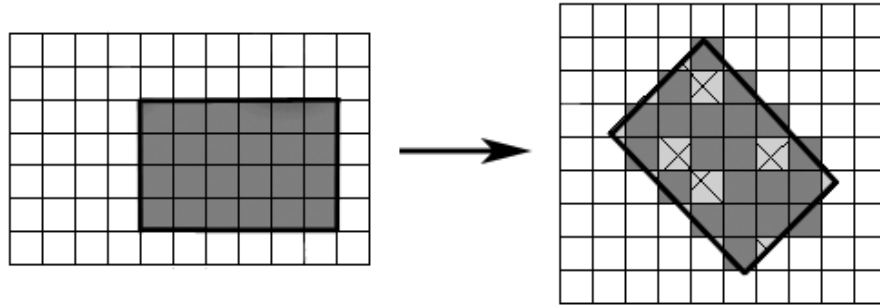
Obrázek 11: Transformace mezi kartézskými systémy ve 2-D, zdroj vlastní

2.6 Geometrické transformace diskrétního obrazu

Principy geometrických transformací popsaných v předchozích kapitolách předpokládají reprezentaci obrazu fyzickým pixelem nebo spojitou funkcí. Diskrétní obraz na výstupu je však už jiné povahy. Při samotném vykreslování obrazu nastávají jisté problémy a je třeba do geometrických transformací zakomponovat numerické metody pro kvalitní reprezentaci transformovaného obrazu.

Obecná geometrická transformace způsobí, že se diskrétní souřadnice $[i; j]$ vstupního pixelu zobrazí na celočíselné souřadnice. Po provedení zpětného přechodu k celočíselné souřadnici pixelu $[x; y]$ tedy budou vznikat mezery nebo se některé pixely budou při mapování překrývat. Situaci po rotaci a mapování do celočíselných souřadnic ilustruje obrázek 12. Mapování určuje způsob přiřazení pixelů z jednoho obrazu do druhého. Odlišují se dva způsoby mapování, a to dopředné (*forward mapping*) a zpětné mapování (*backward mapping*). Při dopředném mapování procházíme pixely vstupního obrazu a hledáme jejich umístění ve výstupním obrazu. Při zpětném mapování naopak procházíme pixely výstupního obrazu a hledáme jim odpovídající oblasti v obrazu vstupním. Geometrické

transformace, které dovolují změny velikosti obrazu, komplikují průběh těchto mapování. [Žára04]



Obrázek 12: Diskrétní transformace rotace, zdroj vlastní

Při proudovém zpracovávání grafické informace jsou s výhodou využívány **separabilní operace**. Transformace je separabilní, pokud ji lze vyjádřit jako složení dvou transformací, tj. $T(u, v) = F(u, G(v))$. Například první transformace G pracuje pouze se řádky obrazu, zatímco transformace F transformuje sloupce. Typickou separabilní operací jsou dvouprůchodové operace otáčení. [Žára04]

V následujícím textu budou stručně popsány vybrané metody, které pracují na principu zpětného mapování. Širší popis problematiky je nad rámec této bakalářské práce. Je však vhodné naznačit některé souvislosti.

Cílem těchto metod je nalézt přechod od diskrétní funkce I_i ke spojitě funkci $f(x)$. Tato operace se nazývá **rekonstrukce**. Rekonstrukce není jednoznačně určená úloha, protože signál diskrétního obrazu se musí aproximovat. Způsob aproximace určuje různé metody, které se liší kvalitou a rychlostí. [Žára04]

Nejjednodušší metodou rekonstrukce je **interpolace nejbližším sousedem** (*nearest neighbour interpolation, point shift, sample and hold*). Principem této metody je okopírování hodnoty nejbližšího souseda v okolí vzorku. Implementuje se zaokrouhlením reálného čísla. Jedná se o rychlou metodu, která však například při zvětšení zvýrazňuje skoky a při zmenšení poškozuje tenké čáry. Pro svou rychlost je používána k rychlým náhledům. [Žára04] **Bilineární interpolace** používá pro rekonstrukci známé hodnoty čtyř pixelů z okolí bodu, jehož hodnoty hledáme, a proto je metodou rychlou. Bilineární interpolace je operace separabilní. Její nevýhodou je, že rozmazává původní ostré přechody. Tato metoda je často implementována v technickém vybavení počítačů. [Žára04]

Kvalitnějších výsledků je dosaženo interpolačními funkcemi, které mají hladší průběh, neboť lépe respektují ostrost obrazu. Tyto metody potřebují pro svůj výpočet více vzorků ze svého okolí. Tím jsou i výpočetně náročnější. Takto pokročilou metodou je například **kubická interpolace** či aproximace pomocí **kubické B-spline křivky**⁴. [Žára04]

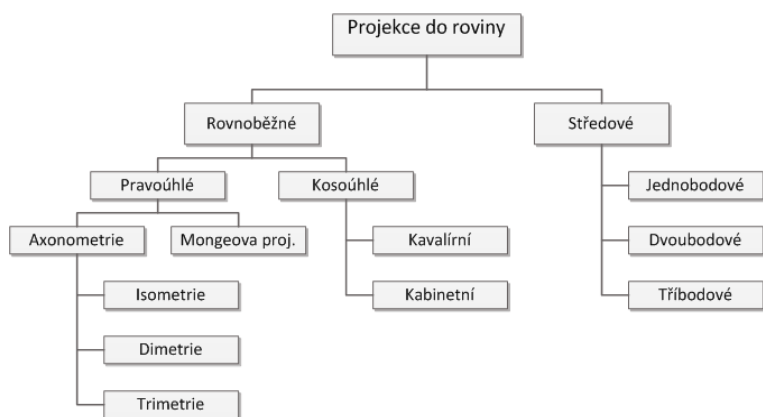
⁴Informace o jednotlivých metodách je možno si doplnit z [Žára04], odkud je čerpán nutný základ pro

Častým požadavkem na transformaci obrazu je **změna rozlišení** obrazu. Různé algoritmy pro změnu velikosti obrazu se liší zejména tím, jak kvalitně rekonstruují spojitou funkci z diskrétního obrazu. Nejjednodušším případem je zaokrouhlení na hodnotu nejbližšího souseda, složitější algoritmy používají více pixelů z původního obrazu. Jestliže na diskrétní obraz chceme aplikovat transformaci **otočení**, výsledný obraz může zaujímat více pixelů než obraz původní. V tomto případě se pak prázdná místa vyplní barvou pozadí. Rotace o násobky $\pi/2$ takové problémy nevyvolávají. [Žára04]

tuto podkapitolu. Publikace [Watt00] a [Watt92] dále poskytují širší popis tématu v praxi.

3 Projekce z prostoru do roviny

Kapitola 2 pojednávala o metodách pro umístění modelu objektu v rovině nebo prostoru. Nyní bude věnována pozornost tomu, jak je tento model zobrazen. V počítačové grafice jsou trojrozměrné objekty zobrazovány pomocí dvourozměrných zobrazovacích zařízení a transformace, která charakterizuje převod trojrozměrného objektu do dvourozměrné reprezentace, se nazývá **promítání** neboli **projekce**. Při promítání dochází ke ztrátě prostorové informace a tím i k možnému zkreslení názoru pozorovatele na skutečný tvar objektu. Proto jsou pro určité obory vybírány různé způsoby promítání. [Buss03, Žára04] Všechny promítací metody (bez ohledu na jejich specifika) provází pojmy promítací paprsek a průmětna. **Promítací paprsek** je polopřímka procházející promítaným bodem v prostoru, jejíž směr závisí na zvolené promítací metodě. Plocha v prostoru, na kterou dopadají promítací paprsky, se nazývá **průmětna** a na ní vzniká v místě dopadu obraz. Zobrazení objektu ve třírozměrném prostoru je podobné pořízení fotografie, a proto se vžil pro bod, z něhož promítací paprsky vycházejí, pojem **kamera**. [Žára04, Hear97] V dalším textu kapitoly se zaměřím pouze na promítání do rovinné průmětny, jež reprezentují dvě základní třídy - rovnoběžné (paralelní) a středové (perspektivní) projekce. Kromě těchto typů budou zavedena promítání, která již nelze popsat pomocí lineární transformace. Tyto transformace jsou určeny pro speciální aplikace, jako jsou například mapové projekce. Schéma rozdělení základních promítání do roviny znázorňuje obrázek 12. [Žára04, Salo06]



Obrázek 13: Schéma klasifikace promítání do roviny, zdroj vlastní

3.1 Obecný postup konstrukce promítání

Úlohu promítání lze rozložit na tyto kroky:

1. definování souřadných systémů - world coordinate system (WCS), neboli světová soustava souřadnic, a view space (VCS) neboli soustava souřadnic pozorovatele;
2. formulace promítací úlohy, tj. výběr geometrických parametrů, s jejichž pomocí lze jednoznačně určit pozici kamery, průmětny, směru promítání aj.;
3. stanovení transformace, která popíše promítání prostorových bodů do průmětny;
4. nalezení transformace mezi souřadnými systémy WCS a VCS a její vyjádření v maticovém tvaru. [Žára04, Watt00]

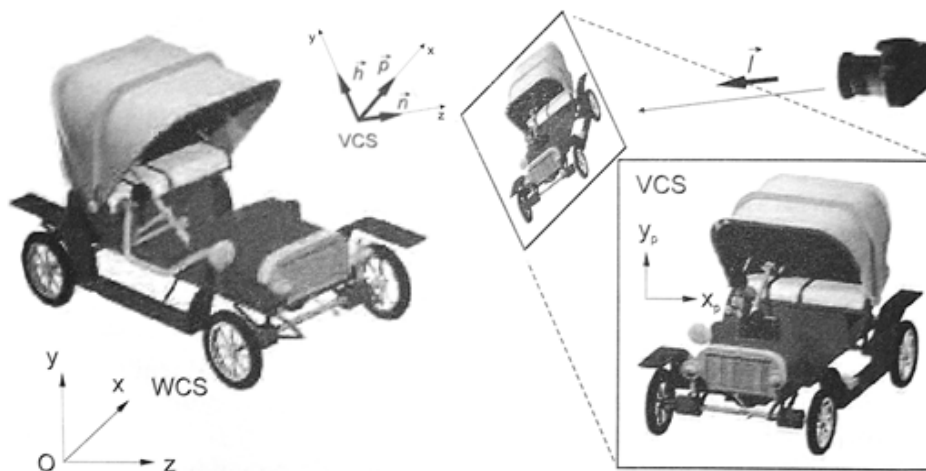
Cílem je nalezení jedné nebo více transformačních matic, které definují transformace souřadnic promítaných bodů podle vztahu

$$\begin{pmatrix} x_p \\ y_p \\ z_p \\ w_p \end{pmatrix} = \mathbf{T}_{proj} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix},$$

kde $P_p = [x_p, y_p, z_p, w_p]$ jsou VCS souřadnice promítaného bodu $[x, y, z, 1]$ v homogenních souřadnicích a $\mathbf{T}_{proj}$ je matice projekce typu 4×4 použitá při transformaci promítání. [Žára04]

Kroky 2 a 3 jsou obsahem následujících podkapitol. Nelze však opominout rozbor kroků 1 a 4. Jak bylo psáno dříve v podkapitole 1.3, WCS je souřadný systém, ve kterém je objekt vymodelován (pomocí geometrických transformací) a VCS je zaveden pro definování promítacích transformací určujících, jak model bude zobrazen. Pro názornost uvádím obrázek 13.

Základem určení VCS je umístění bodu ve WCS, který určí polohu kamery. Ve WCS je nastavena také průmětna. Většinou je požadováno, aby promítací paprsky dopadaly kolmo na průmětnu. Tak je získán vektor $\vec{n} = -\vec{l}$, který je rovnoběžný s hlavní optickou osou kamery a splývající s kladným směrem osy z systému VCS. Orientaci kamery (natočení okolo optické osy) určuje vektor $\vec{h}$, jenž je kolmý na $\vec{n}$, rovnoběžný s průmětnou a především určující kladný směr osy y systému VCS. Z vektorů $\vec{n}$ a $\vec{h}$ je vypočten třetí vektor $\vec{p} = \vec{n} \times \vec{h}$ splývající s kladným směrem osy x . Pro získání transformační matice pro převod WCS do VCS musí být vektory $\vec{n}$, $\vec{h}$ a $\vec{p}$ normovány. Až na posunutí je souřadný systém VCS vztažen ke kameře určen trojicí vzájemně kolmých jednotkových vektorů $\vec{n}$, $\vec{h}$, $\vec{p}$. Předtím, než je objekt promítnut na průmětnu, musí být transformován do VCS



Obrázek 14: Projekce objektu na rovinnou průmětnu, zdroj vlastní, [Žára04]

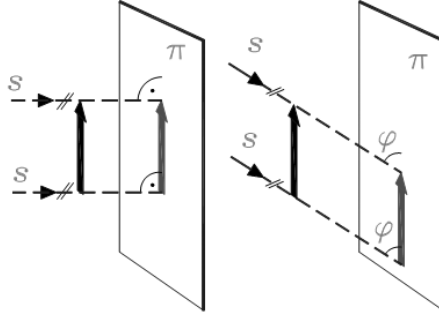
souřadnic. Konverze popisu objektu z WCS do VCS souřadnic probíhá pomocí základních geometrických transformací tak, jak bylo naznačeno v podkapitole 2.5. Posloupnost transformací tedy je: posunout bod kamery do počátku WCS a aplikovat rotaci na souřadné osy VCS tak, aby splynuly s osami WCS. [Žára04, Hear97] Kompletní *WCS-to-VCS* transformační matice je $\mathbf{M}_{w,v} = \mathbf{R} \cdot \mathbf{T}$, kde $\mathbf{T}$ je matice transformace posunutí a

$$\mathbf{R} = \begin{pmatrix} p_x & p_y & p_z & 0 \\ h_x & h_y & h_z & 0 \\ n_x & n_y & n_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

3.2 Rovnoběžné promítání

Rovnoběžné promítání je specifikováno směrem promítacích paprsků, protože kamera je (nevlastním) bodem umístěným v nekonečnu. Jestliže je promítání kolmé na průmětnu, pak je nazýváno **pravoúhlým rovnoběžným promítáním**. V jiném případě, kdy promítací paprsky svírají s průmětnou úhel $\varphi \neq 90^\circ$, jde o **kosoúhlé promítání**. Oba druhy rovnoběžného promítání zobrazuje obrázek 14. [Salo06]

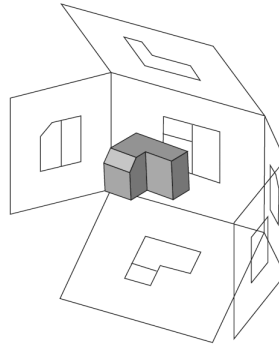
Promítání do některé z rovin $x = x_0, y = y_0, z = z_0$ ve směru příslušné osy kolmé na zvolenou průmětnu je nejjednodušším typem rovnoběžného promítání. Průmětnou může být samozřejmě i některá ze souřadných rovin. Jestliže je $z = z_0$ průmětna, pak libovolný bod $P[x, y, z, 1]$ má v projekci souřadnice $x_p = x, y_p = y$, kde původní souřadnice z je zachována pro řešení hloubky a viditelnosti. Skupina promítání ve směru hlavních os do průměten v hlavních rovinách, která zahrnuje nárys a podle potřeby bokorysy, půdorys, spodní pohled a pohled zezadu (viz obrázek 15), se nazývá **Mongeovo promítání**. [Žára04] Pravoúhlé paralelní promítání se používá v případech, kde je potřeba přesného



Obrázek 15: Pravoúhlé a kosoúhlé promítání, zdroj [Něme09]

měření (CAD, technické výkresy).

$$\mathbf{T}_{xy} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{T}_{xz} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{T}_{yz} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



Obrázek 16: Mongeovo promítání, zdroj [Salo06]

3.2.1 Axonometrie

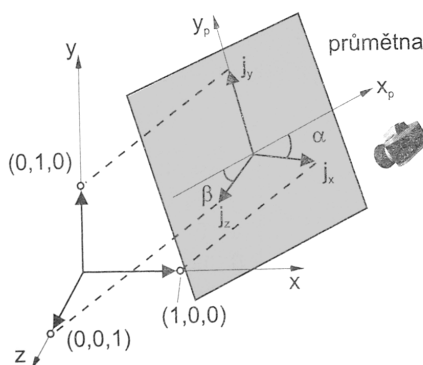
Předchozí pravoúhlé promítání zobrazí detaily objektu pouze z jedné strany, mnohem méně nebo nic však o zbytku objektu. Axonometrie je kompromisem, který promítne tři nebo více stran objektu, ale za cenu zkreslení rozměrů a úhlů. Jsou však zavedeny standardy, které specifikují orientaci objektu vzhledem k pozorovateli a díky nim je možné vzdálenosti z projekce vypočítat. [Salo06]

Axonometrie promítá na průmětnu, která není rovnoběžná se žádnou ze souřadných rovin. Průmětna pak protíná dvě nebo tři hlavní osy WCS. Podle obrázku 16 je axonometrie často definována pěti hodnotami $j_x, j_y, j_z, \alpha, \beta$, kde j_x, j_y, j_z jsou průměty jednotek na osách x, y, z a α, β jsou úhly, které svírají promítnuté osy j_x, j_z s kolmicí na průmět osy j_y .

[Žára04] Transformační matice odpovídající obrázku 16, kde souřadný systém průmětny je zvolen tak, aby průmět j_y ležel na ose y_p , má podobu

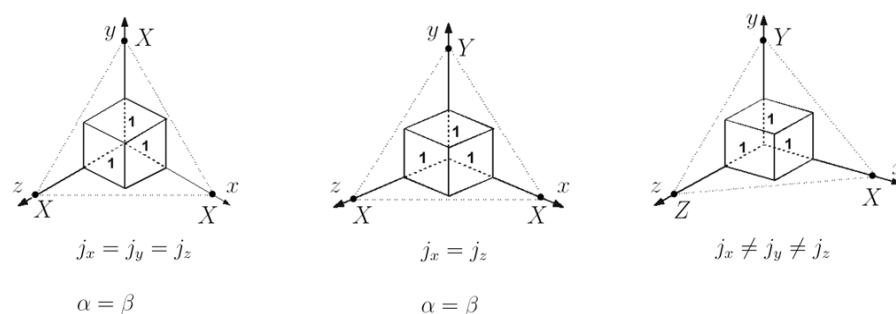
$$\mathbf{T}_{ax} = \begin{pmatrix} j_x \cos \alpha & 0 & -j_z \cos \beta & 0 \\ -j_x \sin \alpha & j_y & -j_z \sin \beta & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

V obecném případě je nutno doplnit transformace otočení a posunutí v rovině průmětny.



Obrázek 17: Zadání axonometrie, zdroj [Žára04]

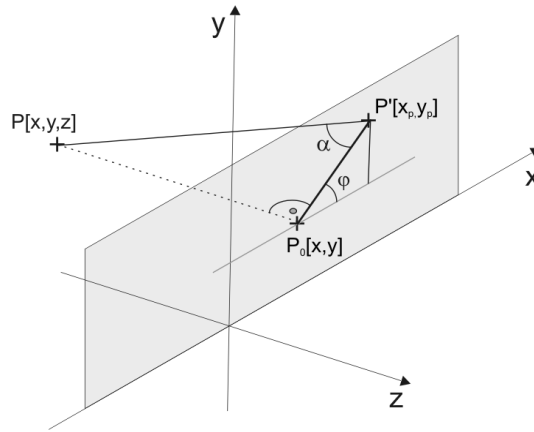
Pokud průmětna protne hlavní osy ve stejné vzdálenosti od počátku WCS (svírá stejný úhel se všemi osami), lze v průmětu měřit a porovnávat vzdálenosti, zkrácení vzdáleností je totiž ve všech směrech promítnutých os stejné. Toto promítání se nazývá **izometrie**. Mají-li shodnou vzdálenost od počátku pouze dva průsečíky, lze v průmětně měřit jen ve dvou směrech promítnutých os, ve třetím směru jsou zkrácené nebo prodloužené vzdálenosti. Tento případ se nazývá **dimetrie**. Při obecném sklonu průmětny vůči osám hovoříme o **trimetrii**, v každém směru promítnuté osy musí být při porovnávání uplatněno odlišné měřítko. Tyto tři třídy axonometrie shrnuje obrázek 17. [Žára04]



Obrázek 18: Izometrie, dimetrie a trimetrie (zleva), zdroj vlastní, [Peli00]

3.2.2 Kosoúhlé promítání

Kosoúhlé promítání kombinuje vlastnosti Mongeova promítání s axonometrickým. Průmětna je rovnoběžná s některou ze souřadných rovin, směr rovnoběžného promítání však není kolmý na průmětnu. Používá se v technické praxi, zejména v architektuře, protože se zvětšující se vzdáleností od pozorovatele nezkracuje vzdálenosti v rovinách rovnoběžných s průmětnou a poskytuje současně boční pohled na promítaný objekt. [Žára04]



Obrázek 19: Kosoúhlé promítání, zdroj vlastní

Pro odvození transformační matice vycházím z obrázku 18, kde bod P je promítaný bod v prostoru a bod P' je obraz v průmětně xy . Pomocí pravoúhlého průmětu P_0 určím souřadnice hledaného bodu P' . Jestliže označím $L = |P_0P'|$, pak

$$x_p = x + L \cdot \cos \varphi, \quad y_p = y + L \cdot \sin \varphi, \quad \text{kde} \quad L = z / \tan \alpha = z \cdot L_1.$$

Transformační matice pro rovnoběžné kosoúhlé promítání na rovinu xy má podobu

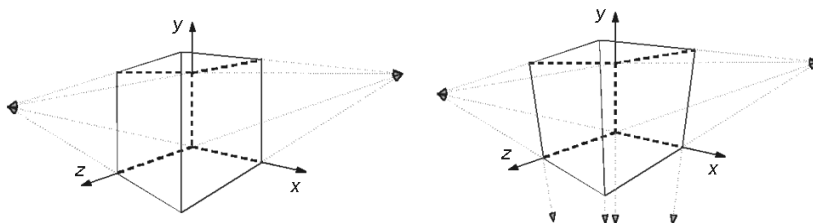
$$\mathbf{K}_{xy} = \begin{pmatrix} 1 & 0 & L_1 \cos \varphi & 0 \\ 0 & 1 & L_1 \sin \varphi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Promítání je pravoúhlé, jestliže $L_1 = 0$, a tedy $\alpha = 90^\circ$. Běžnou volbou pro úhel α jsou hodnoty, pro které platí $\tan \alpha = 1$ a $\tan \alpha = 2$. V prvním případě, kdy $\alpha = 45^\circ$, se takové promítání nazývá **kavalírní**. Všechny přímky kolmé na průmětnu jsou promítány beze změny svých délek. V druhém případě se jedná o **kabinetní** projekci, kde $\alpha \approx 63,4^\circ$. Přímky kolmé na průmětnu jsou promítnuty v poloviční délce. Úhel φ může však být volen libovolně. [Hear97]

3.3 Středové promítání

Středové (perspektivní) promítání odpovídá optickému modelu, který vyjadřuje lidské vidění reálného světa. Modeluje proporcionální zmenšování předmětů při vzrůstající vzdálenosti od pozorovatele a poskytuje dobrý prostorový vjem na rovině průmětně. Z toho důvodu je ze všech projekcí v počítačové grafice nejvíce aplikováno. [Žára04]

Charakteristickým rysem středového promítání je, že nezachovává rovnoběžnost přímek. Je to dáno tím, že pohledový objem⁵ perspektivního promítání je komolý jehlan. Výjimkou jsou přímky ležící v rovině rovnoběžné s průmětnou. Podle polohy průmětny jsou rozlišeny tři případy středového promítání. Jednobodová perspektiva vzniká, jestliže průmětna protíná jedinou souřadnou osu. Dvoubodová perspektiva vznikne, pokud průmětna protne dvě ze souřadných os, a trojbodová perspektiva je nejobecnějším případem, kdy průmětna protíná všechny tři souřadné osy (viz obrázek 19). [Shre99, Žára04]



Obrázek 20: Dvoubodové a trojbodové středové promítání, zdroj vlastní, [Peli00]

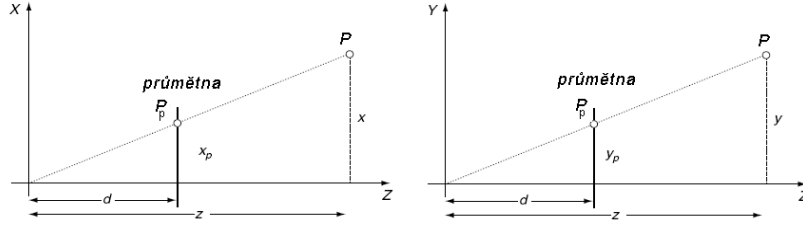
Pro odvození matice středového promítání, kdy střed promítání leží v počátku souřadného systému a průmětna je kolmá na osu z , je využito obrázku 20. Bod $P[x, y, z, 1]$ v homogenních souřadnicích bude transformován na bod $P_p = [x_p, y_p, z_p, w_p]$. Pak z podobnosti trojúhelníků vyplývá, že $x_p = x \cdot \frac{d}{z}$, $y_p = y \cdot \frac{d}{z}$. Vzdálenost d je faktorem změny měřítka souřadnic x a y . Dělení z způsobí projekci vzdálenějších objektů jako menších než objektů bližších středu promítání. Zjevně se nejedná o lineární transformaci. [Fole94, Žára04, Watt00] Projekci však provedeme pomocí matice $\mathbf{T}_{per}$ takto:

$$\begin{pmatrix} x_p \\ y_p \\ z_p \\ w_p \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{d} & 0 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}.$$

Před tím, než je objekt promítnut na výstupní zařízení, je třeba homogenní souřadnice převést. Tato operace se nazývá *perspective division* a spočívá ve vydělení homogenních

⁵Pohledový objem (*viewing volume*) je oblast prostoru ohraničující ty objekty, které mají být podrobeny promítání. Veškeré ostatní objekty musí být před dalším zpracováním odstraněny. [Žára04]

souřadnic číslem w . Zde konkrétně $w = \frac{z}{d}$ a dělením souřadnic tímto číslem jsou získány souřadnice promítnutého bodu. [Watt00]



Obrázek 21: Rozbor projekce zadané situace, zdroj vlastní, [Fole94]

Středové promítání je možno zavést alternativně způsobem, kdy průmětna splývá s rovinou xy a střed promítání leží na ose z ve vzdálenosti d . Podobně jako v předchozím případě jsou souřadnice bodu P_p určeny pomocí podobnosti trojúhelníků. [Fole94] Matice $\mathbf{T}_{per'}$ má tvar

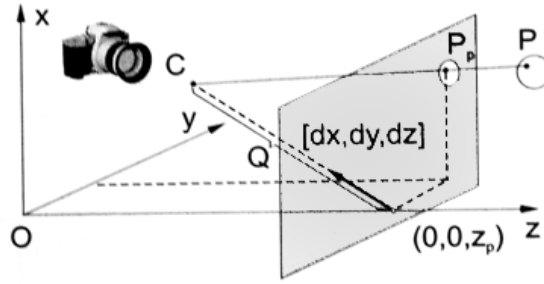
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{d} & 0 \end{pmatrix}.$$

3.4 Jednotné promítání

Rovnoběžné a středové promítání lze popsat jedinou obecnou transformační maticí pro případ, kdy průmětna je kolmá na osu z . Výchozí situaci popisuje obrázek 21. S ohledem na dále odvozené vztahy je předpokládáno, že vektor (d_x, d_y, d_z) je jednotkový. Průmět bodu P je určen z parametrické rovnice přímky $P_p = C + t(P - C)$. Pomocí této rovnice je možno určit libovolný bod $P'(x', y', z')$ na přímce CP , a abychom získali průsečík této přímky s průmětnou, stačí vyjádřit parametr t za podmínky $z' = z_p$. Tedy $t = \frac{z_p - (z_p + Qd_z)}{z - (z_p + Qd_z)}$ a po dosazení do parametrické rovnice a po dalších úpravách získáme souřadnice x_p a y_p . [Fole94, Žára04]

$$x_p = \frac{x - z \frac{d_x}{d_z} + z_p \frac{d_x}{d_z}}{\frac{z_p - z}{Qd_z} + 1}, \quad y_p = \frac{y - z \frac{d_y}{d_z} + z_p \frac{d_y}{d_z}}{\frac{z_p - z}{Qd_z} + 1}, \quad z_p = \frac{-z \frac{z_p}{Qd_z} + \frac{z_p^2 + z_p Qd_z}{Qd_z}}{\frac{z_p - z}{Qd_z} + 1}$$

$$\mathbf{T}_{obecná} = \begin{pmatrix} 1 & 0 & -\frac{d_x}{d_z} & z_p \cdot \frac{d_x}{d_z} \\ 0 & 1 & -\frac{d_y}{d_z} & z_p \cdot \frac{d_y}{d_z} \\ 0 & 0 & -\frac{z_p}{Q \cdot d_z} & \frac{z_p^2}{Q \cdot d_z} + z_p \\ 0 & 0 & -\frac{1}{Q \cdot d_z} & \frac{z_p}{Q \cdot d_z} + 1 \end{pmatrix}$$



Obrázek 22: Rozbor úlohy pro jednotné promítání, zdroj [Žára04]

Z matice $\mathbf{T}_{obecná}$ získáme různé druhy rovnoběžného a perspektivního promítání dosazením příslušných hodnot. Pokud $Q \neq \infty$, zdefinujeme tak jednobodovou perspektivu. [Fole94]

	z_p	Q	d_x	d_y	d_z
$\mathbf{T}_{par}$	0	∞	0	0	-1
$\mathbf{T}_{per}$	d	d	0	0	-1
$\mathbf{T}_{per'}$	0	d	0	0	-1
$\mathbf{T}_{kav}$	0	∞	$\cos \alpha$	$\sin \alpha$	-1

Tabulka 1: Přehled parametrů v jednotném promítání, zdroj [Žára04]

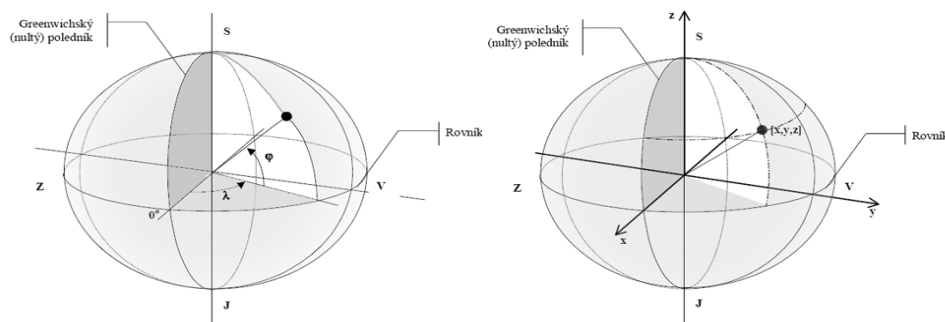
4 Aplikace geometrických transformací v kartografických zobrazeních

Význam geometrických transformací jako primárních operací v počítačovém zobrazování byl osvětlen v podkapitole 1.3. Pro celistvý vhled do této problematiky uvedu souvislosti s konkrétní aplikací geometrických transformací při použití kartografických projekcí, a tedy vytváření map. To, co jednotlivá kartografická zobrazení charakterizuje, je právě souřadný systém a projekce. Zemské těleso je nepravidelné a je tedy nemožno ho přesně matematicky popsat. Zavádí se aproximace zemského povrchu tělesy již matematicky vyjádřitelnými (koule, elipsoid). Z takto zjednodušených modelů je možno povrch Země rozvinout do roviny. Nelze však opominout zakřivení Země. Podle toho, jaké území a s jakou přesností chceme zobrazit, se liší jednotlivé souřadné systémy a projekce. Cílem této kapitoly je ukázat vytváření map v návaznosti na předchozí popsané principy a algoritmy počítačové grafiky.

4.1 Geografická poloha

Jednoznačné definování geometrie objektů na Zemi je možné pouze s použitím vhodného souřadného systému. Pro potřeby geodetického měření a zobrazování potřebujeme ideální tvar zemského tělesa, k jehož povrchu by mohly být body fyzického povrchu Země vztaženy. Ideální zemský povrch lze definovat jako rovnovážnou plochu silového pole zemské tíže. Pokud tato plocha splývá s klidnou střední hladinou moří (prodlouženou i pod pevninu), nazývá se **geoid**. Plocha geoidu ale není matematicky vyjádřitelná. Svým tvarem se přibližuje **rotačnímu elipsoidu**. Je definováno množstvím referenčních elipsoidů, které jsou používány v závislosti na určitém území. Výběr a specifikace referenčního elipsoidu a jeho přesná orientace se označuje jako **datum**. Je snahou zavést jediný společný elipsoid. Mohl by jím být systém WGS (World Geodetic System), kterým se řídí i navigační systém GPS. Pro mnohé úlohy geodézie se v zájmu zjednodušení výpočtu elipsoid nahrazuje **referenční koulí**. [Tuče98]

Polohu bodu na zemském tělese je možno vyjádřit dvěma způsoby. Jestliže je použit **geografický souřadný systém**, pak poloha bodu je dána zeměpisnou šířkou φ (*latitude*) a zeměpisnou délkou λ (*longitude*). Oba parametry jsou měřeny ve stupních. Greenwichskému poledníku odpovídá 0° zeměpisné délky, rovníku odpovídá 0° a pólům 90° zeměpisné šířky. **Kartézský souřadnicový systém** udává polohu kteréhokoli bodu nad i pod povrchem Země pomocí trojice souřadnic $[x; y; z]$. Model Země je proložen kartézskou soustavou s počátkem ve středu Země. Oba modely znázorňuje obrázek 23. [Rapa02]

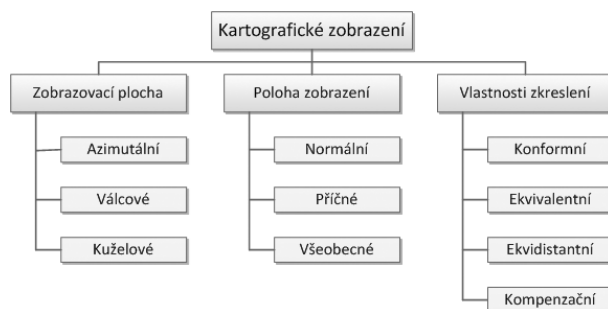


Obrázek 23: Geografický a kartografický souřadný systém (zleva), zdroj vlastní, [Rapa02]

4.2 Kartografická zobrazení

Cílem většiny geodetických prací je zobrazení části zemského povrchu na plochu. Protože je fyzický zemský povrch členitý, je potřeba geodetické prvky (délky, směry, úhly) nejprve redukovat na vhodnou **referenční plochu**, a pak z ní vytvořit zobrazení do roviny rozvinutelné plochy mapy. Referenční plocha je matematicky definovatelná (často jako plocha elipsoidu nebo koule), která se co nejvíce přibližuje geoidu. Za zobrazovací plochu se určí jiné, jednodušší plochy, nejčastěji rozvinutelné do roviny (válcová, kuželová, rovinná). Převod prvků obrazu z referenční plochy do mapy se tedy nazývá **kartografické zobrazení**. [Tuče98]

Kartografických zobrazení je zavedeno velké množství. Obrázek 24 schematizuje jejich základní rozdělení.



Obrázek 24: Schéma rozdělení kartografických zobrazení, zdroj vlastní

- **konformní** (rovnoúhlá) - nezkracují se úhly
- **ekvivalentní** (rovnoplošná) - zachovávají se obsahy ploch
- **ekvidistantní** (rovnodélková) - nezkracuje se určitá soustava linií
- **kompenzační** (vyrovnávací) - obsahuje všechna zkreslení, která jsou stejnou měrou ztlumena

Podle promítacího způsobu a kulové referenční plochy lze získat **gnómicou projekci**, která má střed promítání ve středu koule, **stereografickou projekci**, která má střed promítání v protipólu dotykového bodu zobrazovací plochy a **ortografickou projekci**, když jsou promítací paprsky kolmé na zobrazovací plochu. [Tuče98]

Přehled a vizualizace jednotlivých zobrazení je umístěn v příloze B. Obrázky pro tuto přílohu jsou převzaty z [Kenn00, Komá09].

Tyto projekce jsou koncepčně vytvářeny promítáním jednoho geometrického útvaru (sféry) na jiný (kuželová nebo válcová plocha, rovina). Za účelem snížit zkreslení zobrazení existují mnohé **modifikace** předchozích typů projekcí (např. Mercatorovo prostorové zobrazení v obecné poloze). **Pseudoprojekce** mají určité charakteristiky jiné třídy projekcí. Například Sinusoidální projekce je pseudocylindrickou projekcí, protože všechny poledníky vytnou na rovnoběžkách stejně dlouhé délky, ale na rozdíl od válcové projekce jsou zakřiveny. Výsledná mapa má oválný tvar. [Kenn00] Pro ukázkou je v příloze C uveden výběr několika používaných geografických projekcí. Obrazové a textové informace byly čerpány z [Kenn00, Math10]. Pro další studium o výpočetní realizaci projekcí je vhodný rozsáhlý manuál

4.3 Metody geografických transformací

Přenos geografických dat mezi souřadnými systémy zahrnuje transformaci mezi geografickými souřadnými systémy. Protože datum geografického souřadného systému je založeno na elipsoidu, geografické transformace také základní elipsoid pozměňují. Existuje několik metod, které jsou rozličně přesné a omezené. Geografické transformace vždy konvertují geografické souřadnice. Některé metody ale konvertují geografické souřadnice do kartézských, ty transformují a dále konvertují nové hodnoty do geografických souřadnic. [Kenn00]

4.3.1 Metoda tří parametrů

Nejjednodušší metoda pro transformaci mezi datумы modeluje rozdíl kartézských souřadnic dvou datumů. Transformační parametry jsou obvykle definovány jako výchozí z nějakého lokálního systému do WGS84 nebo jiného geocentrického datumu. Poloha středu nového datumu se nachází ve vzdálenosti $(\Delta X; \Delta Y; \Delta Z)$ od středu původního datumu. [Kenn00]

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix} + \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

4.3.2 Metoda sedmi parametrů

Metoda sedmi parametrů je mnohem komplexnější a přesnější. Sedmi parametry jsou tři lineární posuny ($\Delta X; \Delta Y; \Delta Z$), úhly rotací kolem souřadných os ($\alpha; \beta; \gamma$) a koeficient změny měřítka s . Kladnou hodnotu parametru pro rotaci je možno definovat buď proti směru (USA, Austrálie), nebo po směru hodinových ručiček (Evropa). Pro změnu mezi těmito dvěma systémy stačí změnit znaménka v matici transformace.

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix} + (1 + s) \cdot \begin{pmatrix} 1 & \gamma & -\beta \\ -\gamma & 1 & \alpha \\ \beta & -\alpha & 1 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

4.4 Změna mapové projekce

Využívání vhodného jednotného souřadnicového systému je jedním ze základních předpokladů úspěšné tvorby analýzy a syntézy údajů, jakož i tvorby mapových výstupů. Transformace zeměpisných souřadnic na pravoúhlé rovinné souřadnice jsou **geometrickými transformacemi**. Jejich problematika v kontextu geografie je specifickou částí matematické kartografie. Provedení transformací různého druhu se vykonávají přechodem přes definování polohy na modelu zemského povrchu, tedy na příslušném elipsoidu. Takovéto změny se nazývají **projekce**. [Tuče98]

Mnohdy jsou vstupní mapové podklady vyhotovené v různých kartografických zobrazeních a měřítkách. Pro potřeby různých nástrojů pracujících s geografickými daty je potřeba je převést do jednotného souřadného systému. Význam jednotného souřadného systému roste přímo úměrně s velikostí zpracovávaného území. Na malých vzdálenostech je možno zanedbat zemské zakřivení. [Tuče98]

Transformace hodnot souřadnic mezi pravoúhlými souřadnými systémy nevyžadují znalost zobrazovacích rovnic kartografických transformací. Jsou založeny na poznání přesné polohy vybraných bodů v obou systémech. **Afinní konformní transformace**, tzv. Helmertova transformace, je dána vztahy:

$$\begin{aligned} x' &= (mx \cos \varphi + my \sin \varphi) + a \\ y' &= (-mx \sin \varphi + my \cos \varphi) + b \end{aligned}$$

Souřadné systémy jsou pak vzájemně posunuty, pootočený pro kladnou hodnotu φ po směru hodinových ručiček a se změněným měřítkem ve směru obou os ve stejném poměru. Tato problematika byla podrobně popsána v kapitole 2. Pro optimální odhad parametrů transformace se pro výpočet používá více referenčních bodů a hodnoty koeficientů se stanovují metodou nejmenších čtverců. [Tuče98]

Kromě transformace konformní lze použít **polynomickou transformaci**. Její nejjednodušší formou je polynomická transformace 1. řádu, jejíž zobrazovací rovnice mají tvar:

$$\begin{aligned}x' &= ax + by + c \\y' &= dx + ey + f\end{aligned}$$

Na rozdíl od afinní konformní transformace se jednotlivé souřadnice transformují nezávisle. Korekce ve směru každé souřadné osy se provádí jednotlivě, což je výhodné v případě, kdy materiál mapy je deformován v každém směru jiným způsobem. Pokud má deformace souřadné soustavy mapy komplikovanější průběh nebo lokální charakter, je výhodnější **polynomická transformace vyššího řádu**. Polynomická transformace řádu n má tvar:

$$\begin{aligned}x' &= \sum_{m=0}^n \sum_{i=0}^m a_{m,i} x^i y^{m-i} \\y' &= \sum_{m=0}^n \sum_{i=0}^m b_{m,i} x^i y^{m-i}\end{aligned}$$

Dvojice referenčních bodů je třeba vybírat co nejbližší k okrajům transformovaného území a přesnost transformace zpřesnit body, které se nacházejí uvnitř tohoto území. Aby transformace neměla prudký průběh, rastr se doplní několika body mimo území. V praxi se používají pouze polynomické transformace prvního, druhého a třetího řádu. [Tuče98]

4.5 Převzorkování rastrové reprezentace mapy

Převzorkování (resampling buněk) je definování způsobu přiřazení atributové hodnoty v buňkách výstupního systému. V praxi se tento postup uplatňuje zvláště při sladování údajů z různých zdrojů, při změně souřadného systému nebo při slučování rastrové reprezentace snímku z dálkového průzkumu Země s jinými rastrovými reprezentacemi nebo s mapou. [Tuče98]

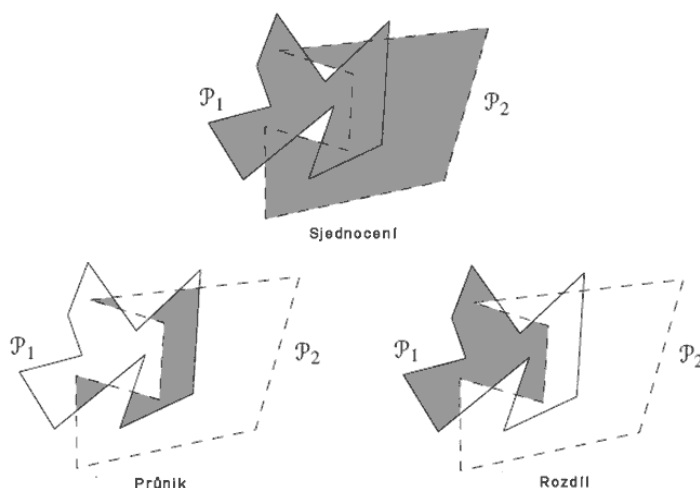
Tento proces je úzce spjat s operacemi transformací diskrétního obrazu, mapováním a rekonstrukcí tak, jak byly naznačeny v podkapitole 2.6. Při převzorkování se podle transformačních rovnic vypočítá poloha středu buňky nové reprezentace v souřadném systému vstupní reprezentace. Po identifikaci polohy středu buňky je možné přiřadit atributovou hodnotu z hodnot uložených v buňkách původní reprezentace. Pro určení přiřazované hodnoty existuje více technik. Přiřazená hodnota při každé technice je jiná a výběr použití je ovlivněn povahou zpracovávaných údajů a účelu vytvářené reprezentace. [Tuče98]

Technika **přiřazení hodnoty nejbližšího souseda** transformované poloze středu výstupní buňky přiřadí atributovou hodnotu z buňky původního rastru, jejíž střed se nachází nejbližší. Při **bilineární interpolaci** se naleznou středy čtyř nejbližších buněk původní reprezentace vůči transformovanému středu výstupní buňky. Přiřazená hodnota je průměrnou hodnotou vah čtyř sousedních vzdáleností. **Kubická interpolace** je podobná

bilineární, ale přiřazená hodnota se počítá jako vážený průměr vzdáleností ze šestnácti nejbližších hodnot vstupní reprezentace. [Tuče98]

4.6 Výpočetní geometrie v geografických informačních systémech

Výpočetní geometrie (*Computational geometry*) je odvětvím počítačové grafiky, která je odlišná od počítačové geometrie, jež byla popisována v předchozích kapitolách. Výpočetní geometrie nebo také geometrické modelování se zabývá především reprezentací objektů reálného světa ve formě odpovídající výpočtům CAD/CAM systémů. Obecnými problémy řešenými výpočetní geometrií je například *Convex hull*, *Line segment intersection*, *lokace bodu*, *triangulace*, *nejkratší euklidovská vzdálenost*, *lineární programování* a další. Všechny tyto problémy zasahují do aplikace geografických informačních systémů, které pracují s obrovským množstvím dat. Protože se jedná již o jinou skupinu algoritmů než je cílem popsat v této bakalářské práci, zvědavého čtenáře pouze odkážu na [DeBe08].



Obrázek 25: Mapové překrytí, zdroj [DeBe08]

5 Implementace geometrických transformací v systému Maple

Tato závěrečná kapitola popisuje cestu jak implementovat algoritmy geometrických transformací v programovacím jazyku CAS systému Maple a jejich následnou vizualizaci ve formě dynamické prostorové geometrie dostupné na webových stránkách. Tyto výstupy jsou určeny všem zájemcům o tematiku geometrických transformací a doplnění jejich matematického zápisu názornou vizualizací i ve 3D. Takováto prezentace algoritmů počítačové grafiky zakládajících se na matematických strukturách vede k jejich hlubšímu pochopení do takových základních podrobností, které hrají již svou významnou roli při tak náročných aplikacích jako jsou například transformace mezi souřadnými systémy používanými v geografických informačních systémech.

5.1 Charakteristika systému Maple

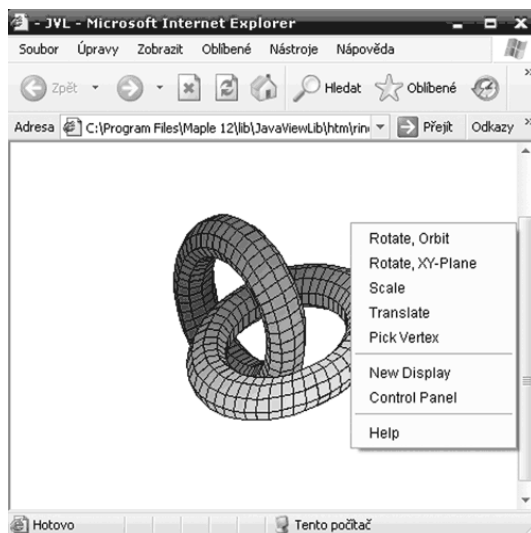
Maple je počítačový systém z řady CAS (Computer Algebra System) programů, které disponují operacemi v symbolické matematice. Umožňuje však i numerické výpočty. Své uplatnění nachází jak ve výuce tak v přírodovědných, technických (mechanika, design strojů, realtime simulace, zpracování signálů) a ekonomických oborech (finanční modelování, operační výzkum). [Mapl10]

V prostředí Maple se pracuje s tzv. hypertextovými zápisníky, ve kterých lze provádět výpočty, vytvářet grafy, jejich animace a doplňovat je texty. Tyto MWS (*Maple Worksheet*) jsou ukládány ve formátu XML dokumentu. V Maple se používá vlastní programovací jazyk čtvrté generace podobný Pascalu s mnoha předdefinovanými funkcemi a procedurami. Soubory MWS lze exportovat do formátu LaTeX, HTML, RTF a nejnovější verze Maple 13 dále umožňuje automatický převod svých příkazů a procedur do programovacích jazyků C, Fortran 77, Java a Visual Basicu. Systém Maple je možno rozšířit o speciální nástroje jako jsou například modelovací a simulační MapleSim, Maple Toolbox for Matlab nebo Maple Financial Modeling Toolbox pro finanční praxi. V systému Maple je možno vytvářet tzv. maplety. Maplet aplikace je grafické uživatelské rozhraní obsahující okna, textová pole a další prvky, které dávají uživateli možnost tzv. *point-and-click* přístupu k nástrojům programu Maple. Uživatelé mohou provádět výpočty, vykreslení grafů a zobrazení dialogu aniž by se pohyboval v prostředí "zápisníku". [Mapl10]

5.2 JavaView

Pro vizualizaci grafických výstupů vytvořených v Maple je výhodné využít balíčku JavaView, a to konkrétně knihovny JavaViewLib určené pro samotný Maple. Tato programová

sada volně stažitelná na oficiálních webových stránkách <http://www.javaview.de/> přináší výhodu nových interaktivních možností pro Maple, protože pomocí ní je možno zabudovat 2D i 3D grafiku do webových stránek, viz obrázek 26. JavaView poskytuje několik možností interakce s grafem pomocí myši jako jsou změna měřítka, posunutí, rotace ve 2D, 3D a 4D, auto-view módy, animace, zbarvení objektu, nanesení textury a průhlednost. Na rozdíl od grafů v Maple jsou grafické prvky hladší a jsou plynuleji animovány. [Java06]



Obrázek 26: Export z Maple pomocí JavaView do html stránky, zdroj vlastní

Protože tematika geometrických transformací a její zpracování v této bakalářské práci je určena pro zájemce o důkladné pochopení základů těchto zobrazovacích metod v počítačové grafice, je třeba poskytnout vizualizaci algoritmů transformací ve formě nezávislé na programovém prostředí Maple. Pomocí JavaViewLib byla tedy vytvořena webová stránka s grafickými výstupy všech procedur.

Je vhodné si uvědomit, že mezi funkcemi, které JavaView nabízí, jsou ve značné míře zastoupeny právě geometrické transformace.

5.3 Procedury a funkce pro geometrické transformace

V prostředí hypertextového zápisníku Maple jsem vytvořila dle zadání bakalářské práce procedury geometrických transformací v rovině a jejich vizualizace. Pro úplnost jsem je doplnila i procedurami pro použití prostorových transformací, jež jsou sice kvůli třetímu rozměru složitější, ale zato však zajímavější. Prostorové transformace jsou založeny na stejných principech jako rovinné, a tedy jejich implementace v daném programovacím jazyku je analogická. Pro programování procedur jsem použila verzi Maple 12.

Procedury a funkce, které jsem vytvořila v programu Maple, jsou uvedeny v sekcích Geometrické transformace v rovině a Geometrické transformace v prostoru. Toto uspořádání

se chová jako soubor kapitol a podkapitol. Podkapitolami pak jsou jednotlivé transformace. Veškerý vytvořený klíčový programový kód je uveden v příloze D.

5.3.1 Geometrické transformace v rovině

Homogenizace2D

Homogenizuje vstupní souřadnice do podoby $[x; y; 1]$.

InvHomogenizace2D

Převede homogenní souřadnice zpět do podoby $[x; y]$.

TRANSFORMUJv2D

Jedná se o obecnou proceduru rovinné transformace, jejímž vstupem je typ transformace, kterou chceme provést. Tato procedura provede homogenizaci, násobení transformačních matic a inverzi k homogenizaci. Pomocí příkazu PLOT vykreslí obraz i vzor.

Objekty

Pro ukázkou transformací bylo třeba vytvořit obraz, se kterým se dále mělo pracovat. Pro jednoduchost jsem zadefinovala trojúhelník a pro zajímavost a souvislost s kapitolou 4 jsem získala rovinné souřadnice hranice ostrova Kypru. Tato data byla extrahována pomocí funkcí, které nabízí program Mathematica 7. Zobrazení územního celku ostrova je v Mercatorově projekci.

Translace2D

Translace je jednoduchou funkcí, kdy vstupem je vektor posunutí a výstupem matice. Sekce obsahuje funkci pro animaci translace, ve které se obraz pohybuje po přímce z polohy vzoru do polohy obrazu.

Rotace2D

Vstupem funkce pro rotaci okolo počátku je pouze úhel otočení. Následně je vytvořena procedura pro rotaci kolem obecného bodu, ve které je zřetelně vidět operace skládání transformací. Nechybí zde ani funkce pro animaci vzoru o daný úhel.

Souměrnosti

Středová souměrnost vzhledem k počátku je konstantní maticí. Tato základní souměrnost je využita v proceduře obecné středové souměrnosti. Následují transformace osové souměrnosti podle osy x a y . Vstupem této procedury jsou konstanty k a q ze směrnice tvaru osy. Protože jsem chtěla ve výsledku zobrazit i samotnou osu, bylo třeba vytvořit proceduru *KresliOsSoum*, která je obohacena o výpočet krajních bodů osy podle polohy vzoru a obrazu. Jako pokročilou podobu animace osové souměrnosti jsem řešila úlohu nalezení obrazu kružnice v osové souměrnosti na základě pohybu bodu.

Meritko2D

Tuto funkci charakterizují dva koeficienty změny měřítka. Opět transformaci doplňuje její animace. Zde jsem použila objektu ostrova Kypru pro další názornost změny mě-

řítka. Vzhledem k povaze dat ostrova jsem pro lepší zobrazení transformaci zkombinovala s posunutím.

Zkoseni2D

Tyto transformace opět tvoří pouze její parametry a tvar transformační matice. Pro lepší představu jsem uvedla více grafických příkladů, protože zvláště v případě skládání zkosení je situace hůře představitelná.

5.3.2 Geometrické transformace v prostoru

Objekty

V případě trojrozměrného prostoru nastává problém jak těleso charakterizovat, aby se po transformaci sestavilo opět jako vzorový objekt. Jelikož jsem se rozhodla provést ukázky transformací na jednotkové krychli s těžištěm v počátku, musela jsem si vytvořit proceduru *SlozStenyKrychle*, která zachovává posloupnost vykreslování polygonů, z nichž je krychle složena. Jako druhý objekt jsem využila dat uložených v knihovně PlotTools, ze které jsem si vyexportovala vrcholy desetistěnu. Na něj jsem v procedurách transformací aplikovala funkci *SlozStenyDesetistenu*.

Homogenizace3D, InvHomogenizace3D

Tyto funkce jsou analogiemi Homogenizace2D a InvHomogenizace2D.

TRANSFORMUJv3Dkrychle, TRANSFORMUJv3Ddes

Na rozdíl od obecné transformace v rovině je zde přidána volba, zda ve výsledném obrázku chceme zobrazit i vzor. Mnohdy se může stát, že by se objekty mohly z části pohlcovat.

Translace3D

Funkce pro transformaci translace a její animaci je pouhou analogií Translace2D.

Rotace3D

Průběh transformace rotace se v různých publikacích věnovaných počítačové geometrii realizuje různorodým způsobem. Jednotlivá řešení se liší vstupem nebo požadavkem na rychlost. Pro implementaci rotace v Maple jsem zvolila algoritmus, který je popsán v příloze A. Pro rotaci kolem libovolné osy je potřeba definovat transformační matice rotací kolem souřadných os.

Souměrnosti

Byla vytvořena funkce středové souměrnosti podle libovolného bodu. Případů osové souměrnosti v prostoru bylo rozlišeno šest (osové souměrnosti podle souřadných os a podle souřadných rovin). Jedná se o jednoduché transformace, proto jsem jejich vizualizaci ukázala ve složení s jinými transformacemi dále.

Meritko3D

Funkce pro transformaci změny měřítka a její animace je pouhou analogií Meritko2D.

Zkos3D

Byly nadefinovány tři funkce pro zkosení podle jednotlivých souřadných rovin. K nim byla vytvořena i funkce pro animaci této transformace.

5.3.3 Knihovny a funkce programu Maple

Při tvorbě procedur jsem využila mnohé předdefinované funkce, které Maple nabízí. Tato podkapitola uvádí výčet těchto funkcí v pořadí jejich použití s jejich stručnou charakteristikou.

Knihovny:

- *with(LinearAlgebra)*: funkce pro operace s maticemi, vektory a jejich dimenzemi
- *with(ListTools)*: práce se seznamy, např. připojení, seřazení, vytvoření posloupnosti
- *with(plots)*: funkce pokročilých zobrazování, animace
- *with(PlotTools)*: soubor speciálních geometrických objektů

Funkce:

- *seq*: posloupnost
- *Dimensions*, *RowDimension*: dimenze (řádková dimenze) matice
- *Matrix*: matice
- *evalf*: zaokrouhlení
- *Transpose*: transponování matice
- *PLOT*, *PLOT3D*: zobrazování rovinných (prostorových) objektů
- *Flatten*: zrušení struktury seznamu do zadané úrovně
- *min*, *max*: minimum (maximum) z posloupnosti
- *Vector*: vektor
- *if-then-else*: struktura podmíněného příkazu
- *sqrt*: odmocnění
- *abs*: absolutní hodnota
- *arcsin*, *arccos*: cyklometrické funkce $\arcsin(x)$, $\arccos(x)$

5.4 Animace v programu Maple

Animace v Maple vzniká na základě vytvoření posloupnosti obrázků (*frames*) v závislosti na proměnné, jež se mění s časem. K tomu slouží příkaz *animate* z knihovny *plots*. Jeho vstupními parametry jsou postupně název procedury dávající grafický výstup, její vstupní proměnné a rozsah změny hodnoty animační proměnné. Lze tedy velmi snadno animovat i přímo příkazy PLOT a PLOT3D pro vykreslování grafických objektů. Tomuto tématu je ve vybraných příkladech věnován článek v časopisu Media4U⁶, jehož jsem spoluautorkou v rámci zpracování tématu této bakalářské práce.

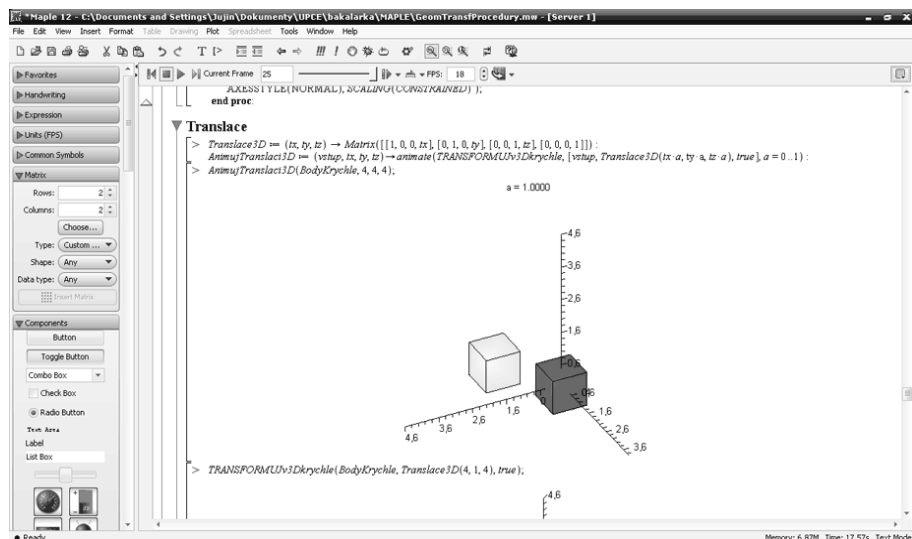
Při realizaci animace v příloze D této bakalářské práce se například používá příkaz v následující formě:

```
AnimujTranslaci3D := (vstup, tx, ty, tz) -> animate(TRANSFORMUJv3Dkrychle,
[vstup, Translace3D(tx*a, ty*a, tz*a), true], a = 0 .. 1);
```

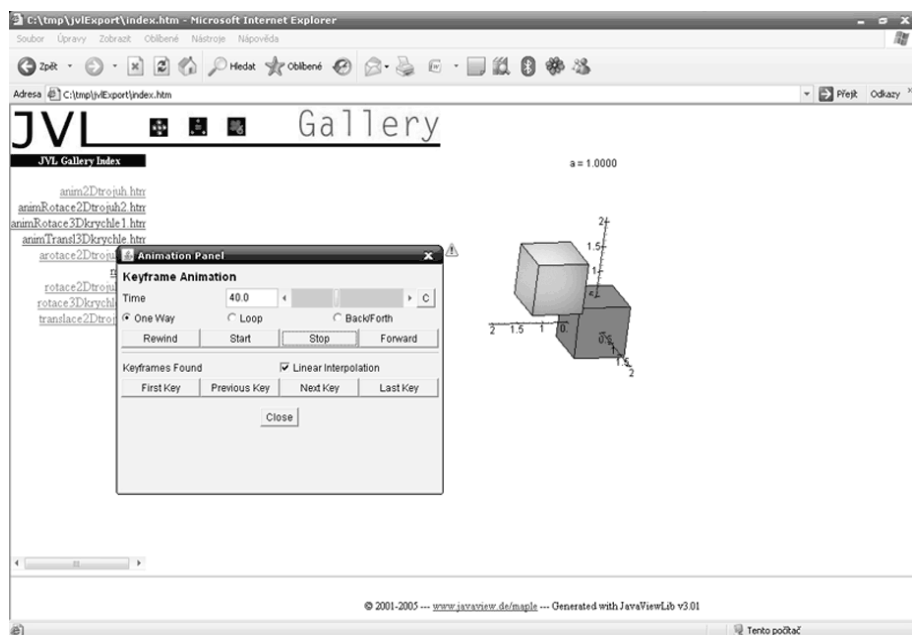
5.5 Grafický výstup procedur pomocí JavaView

Pro využití možností, jež JavaView nabízí, bylo třeba do souboru *mw* tuto knihovnu načíst. Nastavila jsem automatické otevírání webové stránky s vytvořenou grafikou v prohlížeči Firefox a cíl exportů do adresáře `c:\tmp\jvlExport`. Samotný export do interního datového formátu Maple *MPL*, se kterým je JavaView kompatibilní, se provede jednoduchým příkazem `exportMPL(graf):.` Tuto operaci je možno provést najednou s exportem do HTML, a to příkazem `exportHTM(graf, "graf.mpl"):`. Všechny vygenerované HTML stránky lze pomocí příkazu `viewGallery()` shrnout do jedné webové prezentace. Následující obrázek 27 ilustruje prostředí programu Maple s vytvořenou procedurou pro animaci translace v prostoru. Obrázek 28 již ukazuje podobu výstupu této procedury v internetovém prohlížeči. Protože se jedná o animaci, součástí webové stránky je samostatné okno Java aplikace pro ovládání animace. Toto okno za určitý okamžik zmizí a je možno ho vyvolat klávesovou zkratkou Ctrl+A. Situaci s ovládacím panelem znázorňuje obrázek 28. Kompletní webová stránka se všemi objekty a animacemi je umístěna v příloze na DVD.

⁶Trojovský, P., Hladíková, E. Animace v MAPLE. *Media4U* [online]. s. 62-65, 2010, roč. 7, č. 1. ISSN 1214-9187.



Obrázek 27: Translace v prostředí programu Maple, zdroj vlastní



Obrázek 28: Animace translace v prostředí webové stránky, zdroj vlastní

Závěr

V této bakalářské práci jsem se podrobně věnovala geometrickým transformacím v počítačové grafice. Základem bylo uvedení těchto zobrazovacích metod do komplexního systému zobrazování na počítači. Tím jsem vytyčila význam geometrických transformací v počítačové grafice.

Charakter bakalářské práce je spíše popisný a je určen pro ty čtenáře, kteří chtějí uvědomlele pracovat s pokročilými geometrickými transformacemi například v oboru kartografie. Podává veškerý základ pro pochopení průběhu algoritmů geometrických transformací. Jádrem je tedy kapitola 2, jež popisuje jednotlivé geometrické transformace v rovině a prostoru. Jsou zde odvozeny transformační matice a text je doplněn názornými obrázky. V případě rotace v prostoru však vyvstávají určité problémy, protože transformace složením rotací kolem jednotlivých souřadných os je výpočetně náročné. Tento problém je řešen různými aproximacemi nebo aplikováním kvaternionů.

Projekce jako speciální geometrické transformace jsou zavedeny ve třetí kapitole. Je to však tematika široká a velmi zajímavá a popis jejich užití kvůli rozsahu práce není úplný. Projekcím se dále věnuje následující kapitola, jež pojednává o kartografických zobrazeních. Úplné základy kartografických projekcí stojí na principech popsanych v předcházejících kapitolách. Porozumění v praxi používaným projekcím ale vyžaduje hlubší matematické znalosti. Pro ukázkou jsou v příloze uvedeny vybraná zobrazení a jejich zobrazovací rovnice, které jsou často velmi složité.

Praktickou ukázkou algoritmů geometrických transformací je implementace v programovacím jazyku CAS programu Maple. Tento program standardně dovoluje operace s maticemi, vykreslení objektů a animace i v prostoru. Abych zajistila dostupnost výstupů z mnou vytvořených procedur a funkcí, propojila jsem funkce Maple s knihovnou JavaView, jež poskytuje možnost uživateli pracovat s grafickými objekty v prostředí webové stránky.

Seznam použitých informačních zdrojů

- [Buss03] BUSS, Samuel R. *3-D Computer Graphics : A Mathematical Introduction with OpenGL*. 1st edition. Cambridge University Press, 2003. 371 s. ISBN 978-0-521-82103-2.
- [DeBe08] DE BERG, Mark, et al. *Computational Geometry : Algorithms and Applications*. 3rd rev. edition. Springer-Verlag, 2008. 386 s. ISBN 978-3-540-77973-5.
- [Fole94] FOLEY, James D., et al. *Computer graphics - Principles and practice*. 2nd edition. Addison-Wesley, 1997. ISBN 0-201-84840-6.
- [Hear97] HEARN, Donald, BAKER, M. Pauline. *Computer Graphics, C Version*. 2nd edition. Prentice-Hall, 1997. 652 s. ISBN 978-0135309247.
- [Hlav92] HLAVÁČ, Václav, ŠONKA, Milan. *Počítačové vidění*. Grada, 1992. 272 s. ISBN 80-85424-67-3.
- [Java06] *JavaView Homepage* [online]. 1999-2006 [cit. 2010-04-14]. Dostupný z WWW: <<http://www.javaview.de/index.html>>.
- [Kenn00] KENNEDY, Melita. *Understanding Map Projections : GIS by ESRI* [online]. Environmental Systems Research Institute, 1994-2000 [cit. 2010-04-05]. Dostupný z <[http://www.duc.auburn.edu/academic/classes/fory/7470/lab08/understanding map projections.pdf](http://www.duc.auburn.edu/academic/classes/fory/7470/lab08/understanding%20map%20projections.pdf)>.
- [Klet04] KLETTE, Reinhard, ROSENFELD, Azriel. *Digital geometry : Geometric Methods for Digital Picture Analysis*. 1st edition. Morgan Kaufmann Publishers Inc., 2004. 672 s. ISBN 1-55860-861-3.
- [Komá09] KOMÁRKOVÁ, Jitka. *Geographic Information Systems - GIS : Prezentace k výuce předmětu PGIS1, KGIS1*. FES UPCE, 2009 [cit. 2010-04-05].
- [Mapl10] *Maplesoft : Math Software for Engineers, Educators and Students* [online]. 2010 [cit. 2010-04-10]. Dostupný z WWW: <<http://maplesoft.com/>>.
- [Mart02] MARTIŠEK, Dalibor. *Matematické principy grafických systémů*. 1. vyd. Littera, 2002. 278 s. ISBN 80-85763-19-2.

- [Math10] *Map Projections - from Wolfram Mathworld* [online]. 1999-2010 [cit. 2010-04-11]. Dostupné z WWW: <<http://mathworld.wolfram.com/topics/MapProjections.html>>.
- [Něme09] NĚMEC, Martin. *Přednášky do předmětu Základy počítačové grafiky* [online]. 2007-2009 [cit. 2009-11-16]. Dostupný z WWW: <<http://barborka.vsb.cz/nemec/zpg/prednasky/prednaska02.pps>>.
- [Nvid09] *NVidia - World Leader in Visual Computing Technologies* [online]. 2009 [cit. 2009-11-05]. Dostupný z WWW: <<http://www.nvidia.com/>>.
- [Peli00] PELIKÁN, Josef. *Lineární transformace a promítání* [online]. 2000 [cit. 2009-11-17]. Dostupný z WWW: <<http://www.fi.muni.cz/usr/sochor/M4730/TransProm/>>.
- [Rapa02] RAPANT, Petr. Úvod do geografických informačních systémů [online]. VŠB-TU Ostrava, 2002 [cit. 2010-04-05]. Dostupné z WWW: <http://gis.vsb.cz/dokumenty/ugis_i>.
- [Salo06] SALOMON, David. *Transformations and Projections in Computer Graphics*. 1st edition. Springer-Verlag, 2006. 289 s. ISBN 978-1-84628-392-5.
- [Shre99] SHREINER, Dave. *OpenGL Programming Guide* [online]. 1999 [cit. 2009-11-20]. Dostupný z WWW: <<http://www.glprogramming.com/red/index.html>>.
- [Tuče98] TUČEK, Ján. *Geografické informační systémy : Principy a praxe*. Computer Press, 1998. 424 s. ISBN 80-7226-091-X.
- [Snyd87] SNYDER, J. P. *Map Projections - A Working Manual*. U. S. Geological Survey Professional Paper 1395, 1987. 383 s. ISBN 978-9998605060.
- [Watt00] WATT, Alan. *3D Computer Graphics*. 3rd edition. Addison Wesley, 2000. 592 s. ISBN 978-0-201-39855-7.
- [Watt92] WATT, Alan, WATT, Mark. *Advanced animation and rendering techniques : Theory and Practice*. Addison Wesley, 1992. 472 s. ISBN 978-0-201-5412-1.
- [Žára04] ŽÁRA, Jiří, et al. *Moderní počítačová grafika*. 2. přepracované, rozšířené vyd. Computer Press, 2004. 609 s. ISBN 80-251-0454-0.

Seznam obrázků

1	Fyzický a logický pixel	11
2	Valuace formátu GIF	11
3	Zobrazovací řetězec	13
4	Souřadné systémy	13
5	Levotočivý a pravotočivý systém souřadnic	15
6	Rotace kolem obecného bodu - skládání transformací	17
7	Posunutí ve 2-D	17
8	Otáčení ve 2-D	18
9	Středová souměrnost ve 2-D podle počátku a její matice transformace	19
10	Osové souměrnosti ve 2-D a jejich matice transformace	20
11	Transformace mezi kartézskými systémy ve 2-D	25
12	Diskrétní transformace rotace	26
13	Schéma klasifikace promítání do roviny	28
14	Projekce objektu na rovinnou průmětnu	30
15	Pravoúhlé a kosoúhlé promítání	31
16	Mongeovo promítání	31
17	Zadání axonometrie	32
18	Izometrie, dimetrie a trimetrie	32
19	Kosoúhlé promítání	33
20	Dvoubodové a trojbodové středové promítání	34
21	Rozbor projekce zadané situace	35
22	Rozbor úlohy pro jednotné promítání	36
23	Geografický a kartografický souřadný systém (zleva)	38
24	Schéma rozdělení kartografických zobrazení	38
25	Mapové překrytí	42
26	Export z Maple pomocí JavaView do html stránky	44
27	Translace v prostředí programu Maple	49
28	Animace translace v prostředí webové stránky	49

Seznam tabulek

1	Přehled parametrů v jednotném promítání	36
---	---	----

Seznam příloh

PŘÍLOHA A	Obecná úloha otočení v prostoru
PŘÍLOHA B	Typy kartografických zobrazení a projekcí
PŘÍLOHA C	Přehled vybraných kartografických projekcí
PŘÍLOHA D	Programový kód klíčových procedur a funkcí
PŘÍLOHA E	DVD-ROM

PŘÍLOHA A

Obecná úloha otočení v prostoru

Transformační matice rotací okolo souřadných os jsou:

$$\mathbf{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{R}_y(\beta) = \begin{pmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

$$\mathbf{R}_z(\gamma) = \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 & 0 \\ \sin \gamma & \cos \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

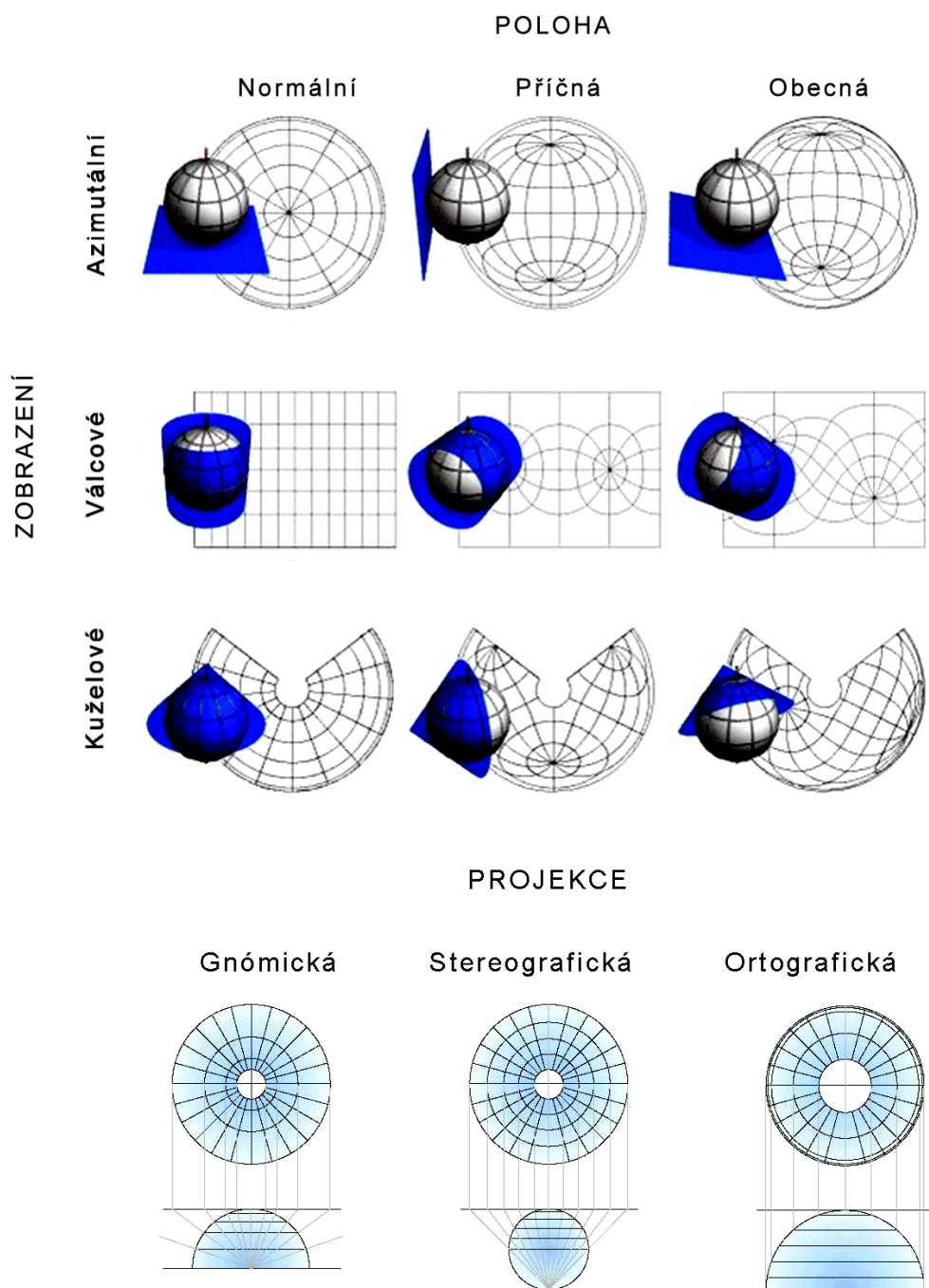
Uvažujme rotaci o úhel φ vzhledem k libovolné ose o , která je zadána parametrickou rovnicí $X = A + p\vec{t}$, $A = [a_1; a_2; a_3]$ a $\vec{t} = (t_x; t_y; t_z)$, přičemž budeme předpokládat, že $|\vec{t}| = 1$.

1. Provedeme postupně převod osy o do osy z :
2. posunutí T_1 s vektorem $\vec{u} = [0; 0; 0] - [a_1 - \frac{a_3}{t_z}t_x; a_2 - \frac{a_3}{t_z}t_y; 0] = (\frac{a_3}{t_z}t_x - a_1; \frac{a_3}{t_z}t_y - a_2; 0)$,
pro $t_z \neq 0$,
3. rotace okolo osy z o úhel α , přičemž $\sin \alpha = \frac{|(0,1,0)(t_x, t_y, t_z)|}{1 \cdot |\vec{t}|} = |t_y|$,
4. rotace okolo osy y o úhel β , přičemž $\cos \beta = \frac{|(0,0,1)(t_x, 0, t_z)|}{1 \cdot |(t_x, 0, t_z)|} = \frac{|t_z|}{\sqrt{t_x^2 + t_z^2}}$,
5. rotace okolo osy z o zadaný úhel φ ,
6. rotace okolo osy y o úhel $-\beta$,
7. rotace okolo osy x o úhel $-\alpha$,
8. posunutí T_2 s vektorem $-\vec{u}$.

Tedy získáme výslednou matici jako součin:

$$T(-a_3 \frac{t_1}{t_2} + a_1, -a_3 \frac{t_2}{t_3} + a_2, 0) \cdot R_x(-\alpha) \cdot R_y(-\beta) \cdot R_z(\varphi) \cdot R_y(\beta) \cdot R_x(\alpha) \cdot T(a_3 \frac{t_1}{t_2} - a_1, a_3 \frac{t_2}{t_3} - a_2, 0).$$

PŘÍLOHA B

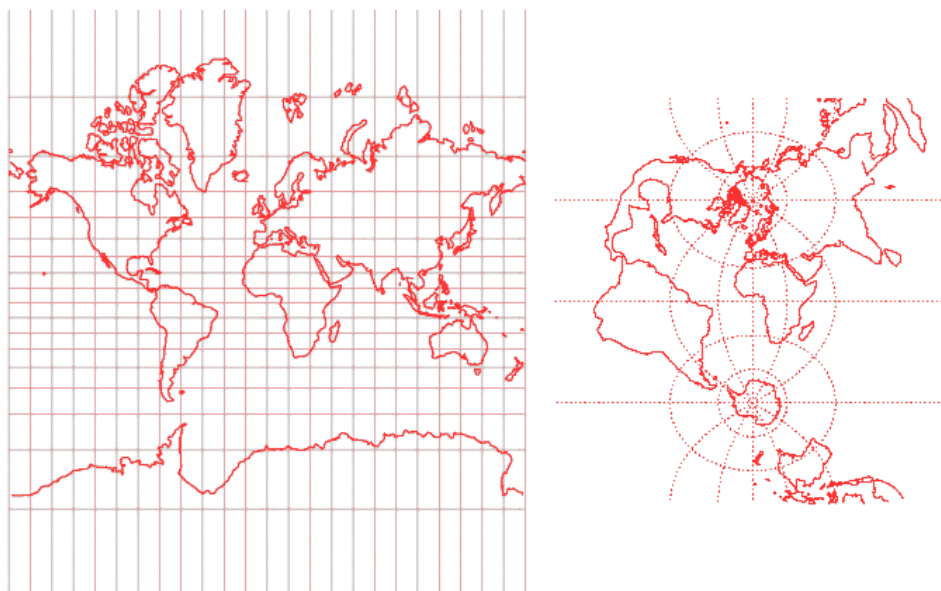


Obrázek 1: Typy kartografických zobrazení a projekcí

PŘÍLOHA C

Mercator projekce

Tato projekce je válcové konformní zobrazení, jež zachovává lokální úhlové vztahy území. Zvětšuje však rozlohu území blíž pólům. Do třídy Mercator projekcí v příčné poloze patří také souřadný systém Gauss-Krügerův a UTM.

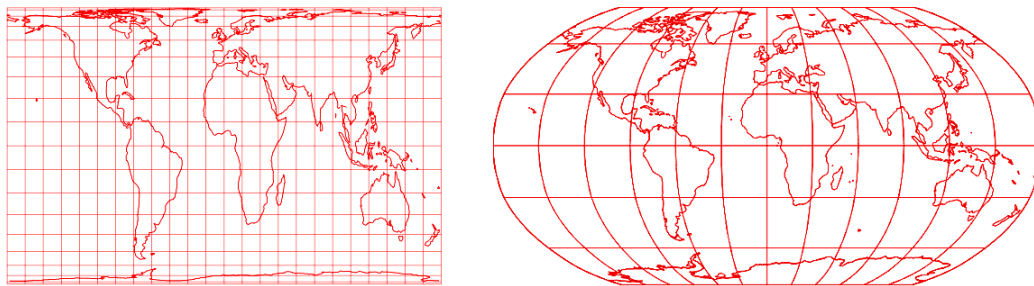


Obrázek 1: Mercator zobrazení v normální a příčné poloze

$$\begin{aligned}x &= \lambda - \lambda_0 & x &= \frac{1}{2} \ln \left(\frac{1+B}{1-B} \right), \text{ kde } B = \cos \varphi \sin(\lambda - \lambda_0) \\y &= \ln \left[\operatorname{tg} \left(\frac{1}{4} \pi + \frac{1}{2} \varphi \right) \right] & y &= \operatorname{arctg} \left[\frac{\operatorname{tg} \varphi}{\cos(\lambda - \lambda_0)} \right] - \varphi_0\end{aligned}$$

Gall-Peters a Robinson projekce

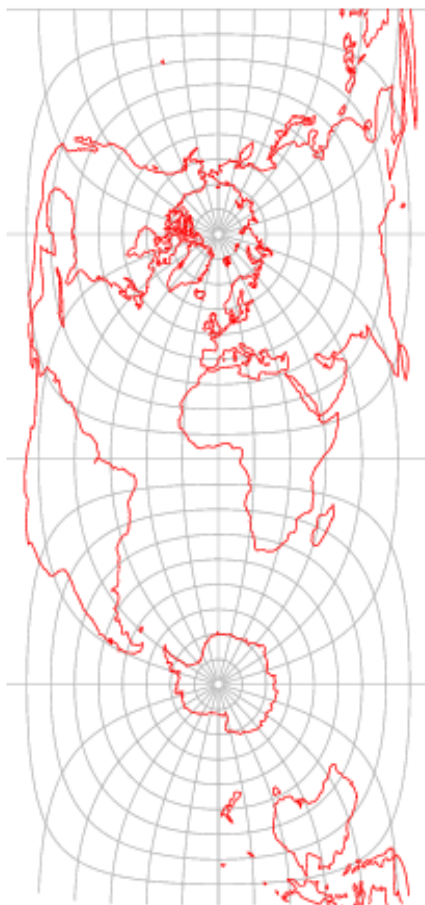
Gall-Peters je válcová perspektivní projekce se základními rovnoběžkami na 45° severní a jižní šířky. Na těchto rovnoběžkách jsou tvary území odpovídající realitě. Zachovává plochu území, ale postupem k pólům i zde vznikají nepřesnosti. Robinsonova projekce je založena na tabulkách souřadnic, ne na matematickém předpisu. Jedná se o pseudo-cylindrickou projekci, jejíž cílem je vytvořit kompromis mezi zkreslením plochy, úhlů, vzdáleností a měřítkem.



Obrázek 2: Gall-Peters a Robinsonova projekce

Cassini-Soldner projekce

Tato projekce je zobrazována pomocí válce v příčné poloze, jenž nezkrsluje území podél nultého poledníku a všech linií s ním rovnoběžných. Střed promítání je v $[0^\circ, 0^\circ]$.



Obrázek 3: Cassini-Soldner projekce

$$\begin{aligned} x &= \arcsin[\cos \varphi \sin(\lambda - \lambda_0)] \\ y &= \arctg\left[\frac{\operatorname{tg} \varphi}{\cos(\lambda - \lambda_0)}\right] \end{aligned}$$

Sinusoidální projekce

Tato pseudocylindrická projekce také zachovává poměr plochy území. Směrem k pólům ale vzniká výrazné zkreslení, naopak útvary kolem nultého poledníku a rovníku jsou méně deformované. Toto zobrazení je výhodné použít, pokud v zájmu stojí například jediný světadíl nebo jiný samostatný útvar.

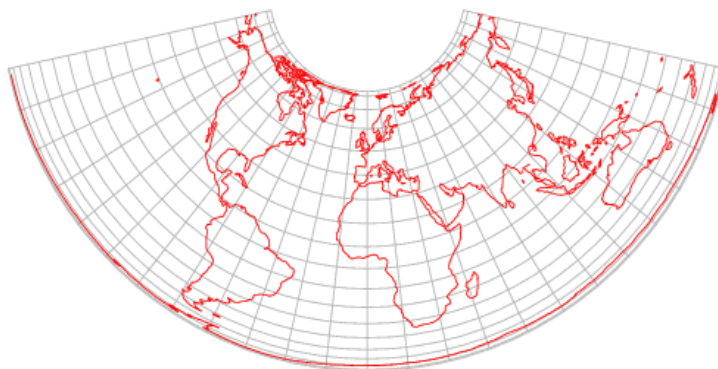


Obrázek 4: Sinusoidální projekce

$$\begin{aligned}x &= (\lambda - \lambda_0) \cos \varphi \\y &= \varphi\end{aligned}$$

Albers projekce

Albers zobrazení je kónická plochojevná projekce, která se řídí dvěma hlavními rovnoběžkami a mezi nimi dochází k minimálním zkreslení. Je vhodné pro zobrazení územních celků, které se rozprostírají v ose východ-západ.



Obrázek 5: Albers projekce

$$x = \rho \sin \theta$$

$$y = \rho_0 - \rho \cos \theta, \text{ kde}$$

$$\theta = n(\lambda - \lambda_0)$$

$$\rho = \frac{\sqrt{C - 2n \sin \varphi}}{n}$$

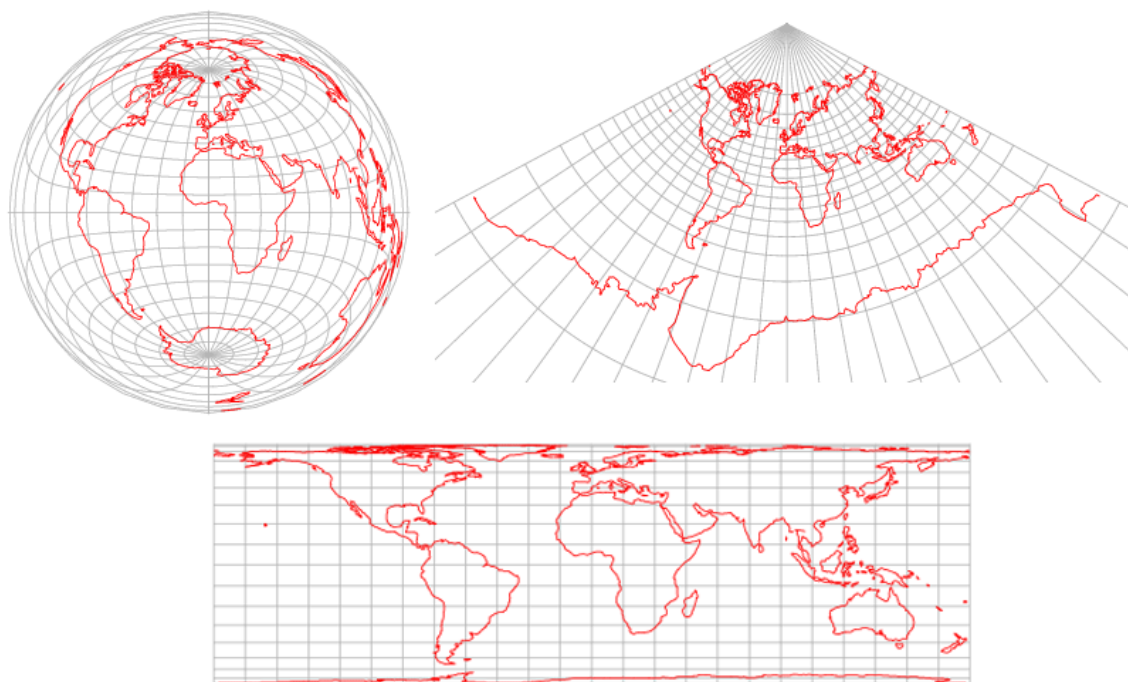
$$\rho_0 = \frac{\sqrt{C - 2n \sin \varphi_0}}{n}$$

$$n = \frac{1}{2}(\sin \varphi_1 + \sin \varphi_2)$$

$$C = \cos^2 \varphi_1 + 2n \sin \varphi_1$$

Lambertovy projekce

Lambertova azimutální projekce zachovává plochu jednotlivých plošných útvarů. Je vhodná pro zobrazení samostatných států. Toto zobrazení může být promítnuto z kteréhokoli bodu. Lambertova konformní kuželová projekce je příbuzná Albersově kuželové s tím rozdílem, že přesněji zobrazuje tvar území než jeho rozlohu. Kužel pro projekci je nastaven v sečnové poloze. Lambertova válcová projekce také zobrazuje zemský povrch tak, že plochy území jsou odpovídající.



Obrázek 6: Lambertova azimutální projekce, Lambertova konformní kuželová projekce, Lambertova válcová projekce

$$\begin{aligned}
x &= k' \cos \varphi \sin(\lambda - \lambda_0) \\
y &= k' [\cos \varphi_1 \sin \varphi - \sin \varphi_1 \cos(\lambda - \lambda_0)], \text{ kde} \\
k' &= \sqrt{\frac{2}{1 + \sin \varphi_1 \sin \varphi + \cos \varphi_1 \cos \varphi \cos(\lambda - \lambda_0)}}
\end{aligned}$$

$$\begin{aligned}
x &= \rho \sin[n(\lambda - \lambda_0)] \\
y &= \rho_0 - \rho \cos[n(\lambda - \lambda_0)], \text{ kde} \\
n &= \frac{\ln(\cos \varphi_1 \sec \varphi_2)}{\ln[\operatorname{tg}(\frac{1}{4}\pi + \frac{1}{2}\varphi_2) \cotg(\frac{1}{4}\pi + \frac{1}{2}\varphi_1)]},
\end{aligned}$$

$$F = \frac{\cos \varphi_1 \operatorname{tg}^n(\frac{1}{4}\pi + \frac{1}{2}\varphi_1)}{n},$$

$$\rho = F \cotg^n \left(\frac{1}{4}\pi + \frac{1}{2}\varphi \right), \quad \rho_0 = F \cotg^n \left(\frac{1}{4}\pi + \frac{1}{2}\varphi_0 \right)$$

PŘÍLOHA D

▼ Geometrické transformace v rovině

```
> with(LinearAlgebra) : with(plots) :  
Homogenizace2D := VB → [seq( [ VB[i, 1], VB[i, 2], 1 ], i = 1  
    ..Dimensions(Matrix(VB)) [1] ) ] :  
InvHomogenizace2D := HVB → [seq( [ HVB[i, 1], HVB[i, 2] ], i = 1  
    ..Dimensions(Matrix(HVB)) [1] ) ] :  
TRANSFORMUJv2D := proc( Vzor, TypTransformace, GrafPrvek := NULL)  
    local Obraz;  
    Obraz := evalf(  
        InvHomogenizace2D(  
            Transpose(Matrix(TypTransformace)).  
            Transpose(Matrix(Homogenizace2D(Vzor)) )  
        )) :  
    PLOT(POLYGONS(Obraz, COLOR( RGB, 0, 0.8, 0.8 ) ), POLYGONS( Vzor,  
        COLOR( RGB, 0.8, 0.2, 0.2 ) ), GrafPrvek, SCALING( CONSTRAINED ) )  
  
end proc;
```

▼ Translace

```
> Translace2D := (tx, ty) → [[ [1, 0, tx], [0, 1, ty], [0, 0, 1] ] :  
    AnimujTranslaci2D := (vstup, tx, ty) → animate( TRANSFORMUJv2D, [vstup,  
        Translace2D( a,  $\frac{ty}{tx} a$  ) ], a = 0 .. tx ) :
```

▼ Rotace

```
> RotaceO2D :=  $\varphi$  → [ [ cos(  $\frac{\varphi \cdot \pi}{180}$  ), -sin(  $\frac{\varphi \cdot \pi}{180}$  ), 0 ], [ sin(  $\frac{\varphi \cdot \pi}{180}$  ),  
    cos(  $\frac{\varphi \cdot \pi}{180}$  ), 0 ], [ 0, 0, 1 ] ] :  
Rotace2D := proc(xr, yr,  $\varphi$ )  
    Matrix( Translace2D(xr, yr) ).Matrix( RotaceO2D(  $\varphi$  ) ).Matrix( Translace2D( -xr,  
        -yr ) )  
end proc;
```

```

AnimujRotaci2D := (vstup, xr, yr, φ) → animate(TRANSFORMUJv2D, [vstup,
Matrix(Rotace2D(xr, yr, F))], F = 0 .. φ, background = PLOT(POINTS([xr,
yr]))):

```

▼ Souměrnosti

```

> StredSoumO2D := [[-1, 0, 0], [0, -1, 0], [0, 0, 1]]:
StredSoum2D := proc(xs, ys)
Matrix(Translace2D(xs, ys)).Matrix(StredSoumO2D).Matrix(Translace2D(-xs,
-ys))
end proc:

> OsSoumX2D := [[1, 0, 0], [0, -1, 0], [0, 0, 1]]:
OsSoumY2D := [[-1, 0, 0], [0, 1, 0], [0, 0, 1]]:
OsSoum2D := proc(k, q)
Matrix(Translace2D(0, q)).Matrix( $\text{RotaceO2D}\left(\frac{\arctan(k) \cdot 180}{\pi}\right)$ )
.Matrix(OsSoumX2D).Matrix( $\text{RotaceO2D}\left(-\frac{\arctan(k) \cdot 180}{\pi}\right)$ )
.Matrix(Translace2D(0, -q))
end proc:

KresliOsSoum := proc(vstup, k, q, GrafPrvek := NULL)
local poslX, Obraz, minX, maxX;
Obraz := evalf(
InvHomogenizace2D(
Transpose(Matrix(OsSoum2D(k, q)).
Transpose(Matrix(Homogenizace2D(vstup))))):

poslX := seq(vstup[i, 1], i = 1 .. RowDimension(Matrix(vstup))), seq(Obraz[i,
1], i = 1 .. RowDimension(Matrix(Obraz)));
minX := min(poslX) - 1; maxX := max(poslX) + 1;
PLOT(POLYGONS(Obraz, COLOR(RGB, 0, 0.8, 0.8)), POLYGONS(vstup,
COLOR(RGB, 0.8, 0.2, 0.2)), CURVES([minX, minX·k + q], [maxX,
maxX·k + q]), GrafPrvek, SCALING(CONSTRAINED))
end proc:

obloukVzor := Array(1..1000): obloukObraz := Array(1..1000): i := 0:
ObrazKruzniceVOsSoum := proc(sx, sy, r, t, k, q)
local vzor, obraz, obrazS, minX, maxX, obrazOblouku; global oblouk, i,
obloukObraz, obloukVzor;
vzor := [r·cos(t) + sx, r·sin(t) + sy];
obraz := Matrix(OsSoum2D(k, q)).Vector([vzor[1], vzor[2], 1]);
obrazS := Matrix(OsSoum2D(k, q)).Vector([sx, sy, 1]);
minX := min(obrazS[1], sx) - r; maxX := max(obrazS[1], sx) + r;

i := i + 1;

```

```

    obloukVzor[i] := vzor, obloukObraz[i] := [obraz[1], obraz[2]];
    PLOT(POINTS(vzor), POINTS([obraz[1], obraz[2]]), CURVES([ [minX, minX
        ·k + q], [maxX, maxX·k + q]]), CURVES([seq(obloukVzor[j], j = 2 ..i)],
        COLOR(RGB, 1, 0, 0)), CURVES([seq(obloukObraz[j], j = 2 ..i)],
        COLOR(RGB, 0.6, 0.1, 0.6)));

```

end proc:

▼ Změna měřítka

```

> Meritko2D := (sx, sy) → [[sx, 0, 0], [0, sy, 0], [0, 0, 1]] :
AnimujMeritko2D := (vstup, sx, sy) → animate(TRANSFORMUJv2D, [vstup,
    Meritko2D(a·sx, sy·a)], a = 0 ..1) :

```

▼ Zkosení

```

> ZkoseniX2D := (shx) → [[1, shx, 0], [0, 1, 0], [0, 0, 1]] :
ZkoseniY2D := (shy) → [[1, 0, 0], [shy, 1, 0], [0, 0, 1]] :
Zkoseni2D := (shx, shy) → [[1, shx, 0], [shy, 1, 0], [0, 0, 1]] :
AnimujZkoseniX2D := (vstup, shx) → animate(TRANSFORMUJv2D, [vstup,
    ZkoseniX2D(a)], a = 0 ..shx) :
AnimujZkoseniY2D := (vstup, shy) → animate(TRANSFORMUJv2D, [vstup,
    ZkoseniY2D(a)], a = 0 ..shy) :
AnimujZkoseni2D := (vstup, shx, shy) → animate(TRANSFORMUJv2D, [vstup,
    Zkoseni2D(a,  $\frac{shy}{shx} \cdot a$ )], a = 0 ..shx) :

```

▼ Geometrické transformace v prostoru

▼ Objekty

```

> SlozStenyKrychle := Body → seq( POLYGONS( [seq(Body[i + 4*j], i = 1
    ..4)], j = 0 ..5 ) :
SlozStenyDesetistenu := Body → seq( POLYGONS( [seq(Body[i + 5*j], i = 1
    ..5)], j = 0 ..11) :

> Homogenizace3D := VB → [seq( [VB[i, 1], VB[i, 2], VB[i, 3], 1], i = 1
    ..Dimensions(Matrix(VB))[1])] :
InvHomogenizace3D := HVB → [seq( [HVB[i, 1], HVB[i, 2], HVB[i, 3]], i = 1
    ..Dimensions(Matrix(HVB))[1])] :

TRANSFORMUJv3Dkrychle := proc (Vzor, TypTransformace, Volba := false,

```



```

    GrafPrvek := NULL)
local Obraz, obrazek;
Obraz := evalf(
    InvHomogenizace3D(
        Transpose(TypTransformace.
            Transpose(Matrix(Homogenizace3D(Vzor)) )
        ))) :
if Volba = true then obrazek := SlozStenyKrychle(Vzor),
    SlozStenyKrychle(Obraz) else obrazek := SlozStenyKrychle(Obraz) end if;
PLOT3D(obrazek, GrafPrvek, LIGHT(0, 0, 0.0, 0.7, 0.0), LIGHT(100, 45, 0.7,
    0.0, 0.0), LIGHT(100, -45, 0.0, 0.0, 0.7), AMBIENTLIGHT(0.4, 0.4, 0.4),
    AXESSTYLE(NORMAL), SCALING(CONSTRAINED) );
end proc:

TRANSFORMUJv3Ddes := proc(Vzor, TypTransformace, Volba := false,
    GrafPrvek := NULL)
local Obraz, obrazek;
Obraz := evalf(
    InvHomogenizace3D(
        Transpose(TypTransformace.
            Transpose(Matrix(Homogenizace3D(Vzor)) )
        ))) :
if Volba = true then obrazek := SlozStenyDesetistenu(Vzor),
    SlozStenyDesetistenu(Obraz) else obrazek := SlozStenyDesetistenu(Obraz)
end if;
PLOT3D(obrazek, GrafPrvek, LIGHT(0, 0, 0.0, 0.7, 0.0), LIGHT(100, 45, 0.7,
    0.0, 0.0), LIGHT(100, -45, 0.0, 0.0, 0.7), AMBIENTLIGHT(0.4, 0.4, 0.4),
    AXESSTYLE(NORMAL), SCALING(CONSTRAINED) );
end proc:

```

▼ Translace

```

> Translace3D := (tx, ty, tz) → Matrix([ [1, 0, 0, tx], [0, 1, 0, ty], [0, 0, 1, tz],
    [0, 0, 0, 1] ]) :
AnimujTranslaci3D := (vstup, tx, ty, tz) → animate(TRANSFORMUJv3Dkrychle,
    [vstup, Translace3D(tx·a, ty·a, tz·a), true], a = 0..1) :

```

▼ Rotace

```

> RotaceX3D := α → evalf( [ [1, 0, 0, 0], [0, cos(α·π/180), -sin(α·π/180), 0], [0,
    sin(α·π/180), cos(α·π/180), 0], [0, 0, 0, 1] ] ) :
RotaceY3D := β → evalf( [ [cos(β·π/180), 0, sin(β·π/180), 0], [0, 1, 0, 0], [

```

```

    -sin( (β·π) / 180 ), 0, cos( (β·π) / 180 ), 0 ], [ 0, 0, 0, 1 ] ] ) :
RotaceZ3D := γ → evalf( [ [ cos( (γ·π) / 180 ), -sin( (γ·π) / 180 ), 0, 0 ], [ sin( (γ·π) / 180 ),
    cos( (γ·π) / 180 ), 0, 0 ], [ 0, 0, 1, 0 ], [ 0, 0, 0, 1 ] ] ) :
Rotace3D := proc( φ, ax, ay, az, t1, t2, t3 )
local t, tx, ty, tz, α, β, MaticeRotace;
t := sqrt( t12 + t22 + t32 );
tx := t1 / t; ty := t2 / t; tz := t3 / t;
α := evalf( abs( arcsin( ty ) · 180 / π ) );
β := evalf( arccos( abs( tz ) / sqrt( tx2 + tz2 ) ) );
MaticeRotace := Matrix( Translace3D( - az·tx / ty + ax, - az·ty / tz + ay, 0 ) )
    .Matrix( RotaceX3D( -α ) ) .Matrix( RotaceY3D( -β ) )
    .Matrix( RotaceZ3D( φ ) ) .Matrix( RotaceY3D( β ) )
    .Matrix( RotaceX3D( α ) ) Matrix( Translace3D( az·tx / ty - ax, az·ty / tz - ay,
    0 ) );
end proc;

AnimujRotaci3D := ( vstup, φ, ax, ay, az, t1, t2, t3 )
    → animate( TRANSFORMUJv3Dkrychle, [ vstup, Matrix( Rotace3D( F, ax,
    ay, az, t1, t2, t3 ) ) ], true, F = 0 .. φ, background = PLOT3D( POLYGONS( [ [ 4,
    3, 2 ], [ 0, -1, -2 ], [ 2, 1, 0 ] ] ) ) ) :

```

▼ Souměrnosti

```

> StredSoumO3D := Matrix( [ [ -1, 0, 0, 0 ], [ 0, -1, 0, 0 ], [ 0, 0, -1, 0 ], [ 0, 0, 0,
    1 ] ] ) :
StredSoum3D := proc( xs, ys, zs )
    Matrix( Translace3D( xs, ys, zs ) ) .Matrix( StredSoumO3D ) .Matrix( Translace3D(
    -xs, -ys, -zs ) );
end proc;

> OsSoumX3D := Matrix( [ [ 1, 0, 0, 0 ], [ 0, -1, 0, 0 ], [ 0, 0, -1, 0 ], [ 0, 0, 0, 1 ] ] ) :
OsSoumY3D := Matrix( [ [ -1, 0, 0, 0 ], [ 0, 1, 0, 0 ], [ 0, 0, -1, 0 ], [ 0, 0, 0, 1 ] ] ) :
OsSoumZ3D := Matrix( [ [ -1, 0, 0, 0 ], [ 0, -1, 0, 0 ], [ 0, 0, 1, 0 ], [ 0, 0, 0, 1 ] ] ) :
OsSoumXY3D := Matrix( [ [ 1, 0, 0, 0 ], [ 0, 1, 0, 0 ], [ 0, 0, -1, 0 ], [ 0, 0, 0, 1 ] ] ) :
OsSoumXZ3D := Matrix( [ [ 1, 0, 0, 0 ], [ 0, -1, 0, 0 ], [ 0, 0, 1, 0 ], [ 0, 0, 0, 1 ] ] ) :
OsSoumYZ3D := Matrix( [ [ -1, 0, 0, 0 ], [ 0, 1, 0, 0 ], [ 0, 0, 1, 0 ], [ 0, 0, 0, 1 ] ] ) :

```

▼ Změna měřítka

> $Meritko3D := (sx, sy, sz) \rightarrow Matrix([[sx, 0, 0, 0], [0, sy, 0, 0], [0, 0, sz, 1], [0, 0, 0, 1]])$:
 $AnimujMeritko3D := (vstup, sx, sy, sz) \rightarrow animate(TRANSFORMUJv3Ddes, [vstup, Meritko3D(sx \cdot a, sy \cdot a, sz \cdot a)], a = 0 .. 1)$:

▼ Zkosení

> $ZkosYZ3D := (shy, shz) \rightarrow Matrix([[1, 0, 0, 0], [shy, 1, 0, 0], [shz, 0, 1, 0], [0, 0, 0, 1]])$:
 $ZkosXZ3D := (shx, shz) \rightarrow Matrix([[1, shx, 0, 0], [0, 1, 0, 0], [0, shz, 1, 0], [0, 0, 0, 1]])$:
 $ZkosXY3D := (shx, shy) \rightarrow Matrix([[1, 0, shx, 0], [0, 1, shy, 0], [0, 0, 1, 0], [0, 0, 0, 1]])$:
 $AnimujZkoseniYZ3D := (vstup, shy, shz) \rightarrow animate(TRANSFORMUJv3Dkrychle, [vstup, ZkosYZ3D(shy \cdot a, shz \cdot a)], a = 0 .. 1)$: