

Warping

Vypracoval: Filip Kotlas

Akademický rok: 2022/23

Studijní program: Matematické inženýrství

Specializace: Matematická informatika

Zadání 1

Zadáním práce bylo vytvořit aplikaci provádějící warping obrazu. Aplikace měla mít grafické uživatelské rozhraní. Měla být schopná načíst obrázky ve standardních formátech JPEG a PNG. Měla umožňovat přidávat, mazat a editovat úsečky z algoritmu. V neposlední řadě měla při výpočtech barvy pixelů používat bilineární interpolaci.

Warping

Warping je druh nelineární transformace obrazu, tj. jeho deformace. Využívá zpětné mapování. Existují dva druhy warpingu, síťový a úsečkový.

První využívá síť, která překrývá původní obrázek. Algoritmus bere řádky pixelů a lineárně přepočítává jednotlivé intervaly mezi čarami sítě. S výsledným obrázkem poté provede stejný proces akorát pro sloupce.

Úsečkový warping používá set párů řídících úseček, kdy jedna z páru je ve zdrojovém a druhá ve výsledném obrazu. Daný bod v transformovaném obrazu je každým párem mapován na vzor tak, že si zachovává svoji relativní polohu vzhledem k těmto úsečkám. Finální bod je pak od svého vzoru posunutý o vážený průměr posunutí pro jednotlivé úsečky zvlášť. Váha je dána vzorcem

$$weight = \left(\frac{(length)^p}{a + dist} \right)^b,$$

kde $length$ je délka dané úsečky v transformovaném obrazu, $dist$ je vzdálenost bodu a úsečky v transformovaném obrazu a a , b a p jsou volitelné parametry.

Bilineární interpolace

Výsledkem popsaného algoritmu je bod ve vzorovém obrazu, který ale prakticky jistě nebude mít celočíselné souřadnice. To se dá řešit buď tím, že souřadnice zaokrouhlíme a vezmeme tím barvu nejbližšího pixelu, nebo použijeme bilineární interpolaci.

Mějme bod s neceločíselnými souřadnicemi. Úkolem je lineárně aproximovat funkční hodnotu v tomto bodě pomocí hodnot v nejbližších bodech s celočíselnými souřadnicemi. Postupuje se tak, že se použije lineární interpolace postupně ve směrech x a y . Tj. označme $f_{x,y}$ interpolovanou funkční hodnotu v bodě (x, y) , kde x, y jsou neceločíselné souřadnice, a

i, j necht' jsou celé části těchto souřadnic. Dále necht' $f_{i,j}$ je funkční hodnota v bodě (i, j) . Pak bilineární interpolace je dána předpisem

$$f_{x,j} = f_{i,j} + (x - i)(f_{i+1,j} - f_{i,j})$$

$$f_{x,j+1} = f_{i,j+1} + (x - i)(f_{i+1,j+1} - f_{i,j+1})$$

$$f_{x,y} = f_{x,j} + (y - j)(f_{x,j+1} - f_{x,j})$$

Zadání 2

Při warpingu čar může docházet k nežádoucím jevům v obraze, způsobených nedostatečným rozlišením obrazu. Takové jevy nazýváme aliasing. Druhým zadáním bylo doplnit aplikaci o funkci analytického antialiasingu, která by tyto jevy potlačila.

Analytický antialiasing

Analytický aliasing provedeme tak, že pomocí warpingu transformujeme na původní obraz celé pixely chápány jako čtverce a následně přes plochu výsledného čtyřúhelníku numericky vyintegrujeme barvu.

K integraci přes transformovaný čtyřúhelník použijeme algoritmus pro vyplňování obecných polygonů. Celý čtyřúhelník překryjeme sítí s rozestupem zadaným integračním krokem. Následně iterujeme přes řádky. Na každém řádku určíme průsečíky s hranami čtyřúhelníku a seřadíme je vzestupně. Do výsledné sumy bereme barevné hodnoty těch čtverců sítě, které leží mezi 1. a 2. a případně 3. a 4. průsečíkem. Nakonec podělíme sumu počtem sčítaných čtverců.

Vypracování

Aplikace byla napsána v jazyce Java za použití knihovny Swing. K jejímu spuštění je proto nezbytné mít instalované Java Runtime Environment a v případě, že je potřeba kompilace i Java Development Kit.

Kompilace a spuštění

Pro kompilování přejděte v shellu na vašem systému do složky se zdrojovými kódy a zadejte příkaz: `javac Warping.java`

Pro spuštění přejděte do složky s bytcodeem. Pokud jste kompilovali pomocí předchozího příkazu, jedná se o tu samou složku. Následně použijte příkaz: `java Warping`

Používání

Po spuštění programu se objeví okno s nabídkou a dvěma zatím prázdnými panely pro obrázky. Než je možné něco upravovat je nutné načíst obrázek. To provedeme kliknutím v panelu nabídky na *Soubor -> Načíst obrázek*. Ukáže se okno, ve kterém budeme moci vybrat soubor ve formátu JPG nebo PNG. Když budeme chtít námi upravený obrázek uložit, klikneme na *Soubor -> Uložit obrázek*.

Pro vložení řídicí úsečky najedeme na jeden z obrázků a stiskneme a podržíme levé tlačítko myši. Následně tažením myši nakreslíme úsečku. Uvolněním tlačítka akci dokončíme. Úsečka se automaticky nakreslí i v druhém obrázku.

Editaci libovolné úsečky provedeme tím, že najedeme myši na jeden z bodů úsečky a zmáčkneme a podržíme levé tlačítko myši. Následně můžeme tento bod přesouvat po obrázku. Uvolněním tlačítka opět akci dokončíme.

Mazání úseček se provádí kliknutím pravým tlačítkem myši na jeden z bodů úsečky. Automaticky se smaže i úsečka v druhém obrázku.

Kliknutím na *Warping* v panelu nabídky se ukážou čtyři položky. Položka *Warping* provede samotný algoritmus a překreslí pravý obrázek. Položka *Nastavení parametrů* vytvoří okno, ve kterém můžeme zadat volitelné parametry pro warping. *Bilineární interpolace* zapne nebo vypne užití bilineární interpolace. Při jejím vypnutí se používá interpolace nejbližším pixelem. *Antialiasing* zapne nebo vypne použití antialiasingu. Antialiasing a bilineární interpolace lze kombinovat.

Poslední v nabídce je položka *Ostatní*, která obsahuje položku *Čáry*, která zapíná a vypíná viditelnost čar v obrázcích, a položku *Nápověda*, která ukáže nápovědu k programu.

Zdroje

1. T. Beier, S. Neely, Feature-based image metamorphosis, Computer graphics (SIGGRAPH '92 Proceedings) (1992), 35-42.
2. https://saint-paul.fjfi.cvut.cz/base/sites/default/files/POGR/POGR1/04.rastrove_algoritmy.pdf
3. https://saint-paul.fjfi.cvut.cz/base/sites/default/files/POGR/POGR1/06.transformace_obrazu.pdf
4. https://saint-paul.fjfi.cvut.cz/base/sites/default/files/POGR/POGR2/02.teorie_signalu.pdf