

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Semestrální práce

Mozaika aneb Co všechno umí pan Voronoi

Obsah

1	Zadání	1
2	Popis použitých algoritmů.....	2
2.1	Algoritmus použitý pro generování náhodně čísel v Gaussovském rozložení	2
2.2	Generování Voronoiovo diagramu	2
3	Popis uživatelského rozhraní aplikace.....	3
3.1	Levá část.....	3
3.2	Pravá část.....	4
3.2.1	Tab sampleru	4
3.2.2	Tab Voronoi.....	5
3.3	Klávesové zkratky.....	5
4	Ukázka vstupního a výstupního obrázku	6
5	Použité zdroje.....	6

1 Zadání

1. Naskenujte si svoji fotografii a uložte v nějakém jednoduchém bitmapovém formátu. Výslednou bitmapu načtěte ve svém programu a náhodně navzorkujte – tj. zjistěte barvu náhodně vybraných pixelů bitmapy. Získanou množinu bodů spolu s informací o barvě uložte do souboru. Umožněte zobrazení takto získané podmnožiny obrázku – buď jako samostatný program anebo v jednom celku se vzorkovacím programem. Pokud vaše řešení skončí tady, získáte **4 body**.
2. Pokud váš vzorkovací program dokáže vzorkovat kromě s rovnoměrným rozložením ještě s normálním rozložením (gauss) a kombinací obou rozložení (tj. rovnoměrně rozložené shluky bodů, např. 10 rovnoměrně rozložených středů normálních rozložení), získáte další **4 body**.
3. Pro takto získanou množinu bodů spočítejte Voronoiův diagram (můžete si napsat vlastní kód anebo použít jakýkoliv legálně dostupný program z Internetu), výsledný diagram obarvěte tak, aby každá Voronoiova buňka měla barvu svého generujícího bodu. Výsledek zobrazte na obrazovce a dovolte opět uložit jako bitmapu. Získaný obrázek by měl vypadat zhruba jako na <http://www.ics.uci.edu/~eppstein/vorpict.html>. Za tuto práci získáte **6 bodů**. Zdůrazňuji, že přípustné je i řešení s velkým podílem ruční práce, tj. jednorázové pracné protlačení vašich dat nějakým programem z Internetu. (V dokumentaci uveďte přesný odkaz.) Není přípustné použít program vašeho spolužáka z VAM.
4. Pokud získaný obrázek (jeden či více) ještě vytisknete na barevné tiskárně a zúčastníte se s obrázkem na cvičení soutěže o nejzajímavější a esteticky nejpůsobivější obraz, získáte další **3 body**. Bude-li dostatečný počet účastníků soutěže, vítězné obrazy budou opět bodově odměněny. (V této soutěži se nehodnotí vaše technické řešení, pouze výsledek.)
5. Jestliže se všechny řešitelské týmy dokáží dohodnout a připravit společně domácí stránku letošního kursu VAM s fotografiemi všech letošních "živých" studentů VAM, kde bude kombinace původních a "upravených" fotografií s nějakým krátkým, vtipným, ale vkusným doprovodným textem, jsou zde další **3 body** pro každý tým.
6. Pokud jste v části 3 vytvořili vlastní programové vybavení a vaše řešení bodů 1 (resp. 2) a 3 bude uživatelsky příjemné a dovolí mi takto zpracovávat bez velkých časových nároků i další bitmapy, získáte dalších **5 bodů**.

2 Popis použitých algoritmů

2.1 Algoritmus použitý pro generování náhodně čísel v Gaussovském rozložení

Algoritmus využívá dvou náhodných čísel z rovnoměrného rozložení. Díky tomu, že máme dvě hodnoty jsme schopni přejít do polárního prostoru a spočítat z již získaných náhodných hodnot hodnoty v Gaussovském rozložení.

```
do
{
    x1 = 2.0 * NextUniform() - 1.0;
    x2 = 2.0 * NextUniform() - 1.0;
    w = x1 * x1 + x2 * x2;
} while (w >= 1.0);

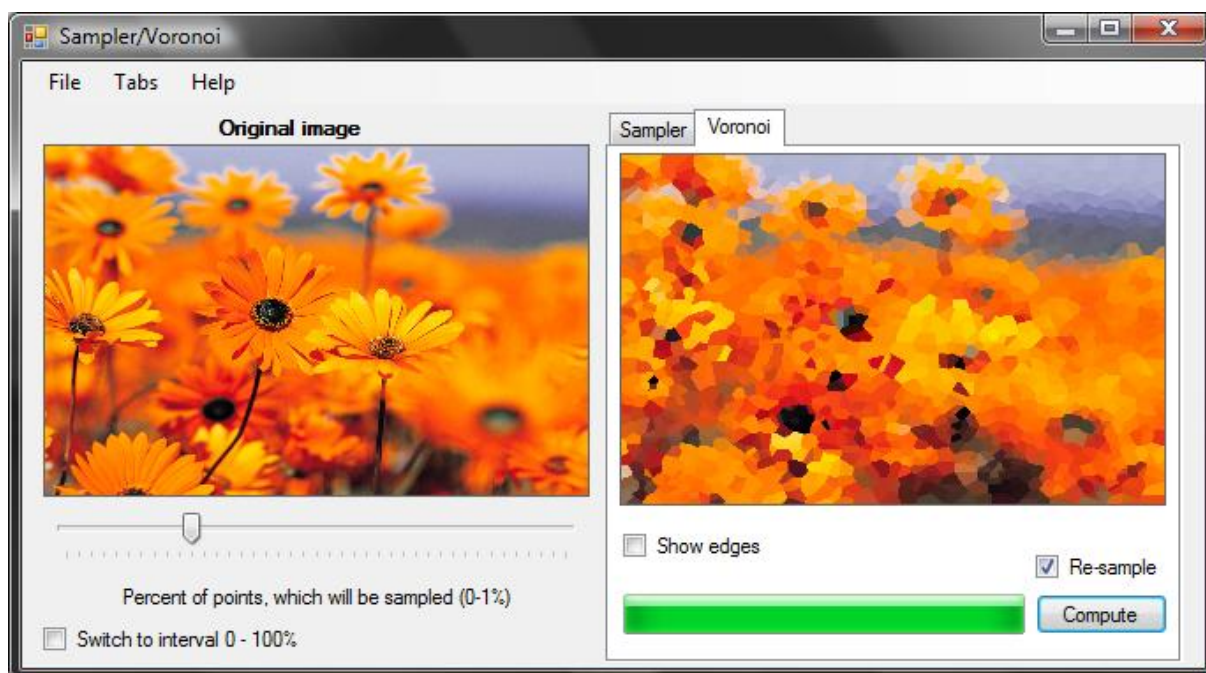
w = Math.Sqrt((-2.0 * Math.Log(w, Math.E)) / w);
y1 = x1 * w;
y2 = x2 * w;
```

2.2 Generování Voronoiovo diagramu

Pro generování Voronoiovo diagramu byla použita knihovna (BenDi), která implementuje tzv. Fortune Algoritmus (Wikipedia). Této knihovně se jako vstup zadávají generující body Voronoiového diagramu a jako výstup lze získat pole hran, které tvoří jednotlivé buňky Voronoiového diagramu.

Z těchto hran je potřeba z rekonstruovat polygony. Jelikož každá hrana nese ukazatel na body generující polygon, provedeme jednoduché rozřazení. Dalším problém, na který lze narazit je ten, že na okrajích jsou polopřímky. Protože každá hrana nese i svůj směrový vektor, je možné dopočítat druhý bod, který použijeme pro vykreslení hrany.

3 Popis uživatelského rozhraní aplikace



Aplikace má dvě základní části. Levá část zobrazuje originální obrázek a v pravé části lze přepínat mezi Samplerem a Voronoiem. V aplikaci je také možné najít menu, kde lze ukládat jak výstup samplera, tak výstup Voronoie. Samozřejmě je možné otevřít vstupní bitmapu.

3.1 Levá část

V této části je možné posuvníkem nastavovat počet bodů, které budou použité pro samplování. Defaultně je rozsah posuvníku nastaven na rozsah 0-1% z celkového počtu pixelů v obrázku. Tento rozsah je ideální pro obrázky běžné velikosti a pro počítání Voronoiovo diagramu.

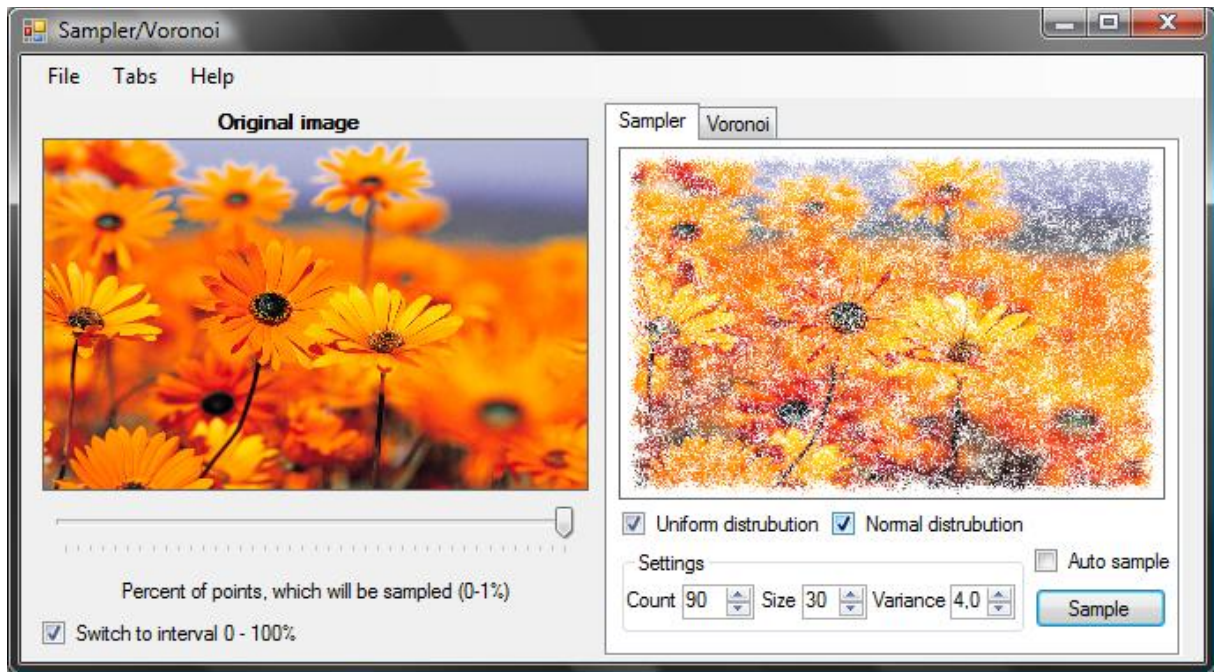
Rozsah je možné přepnout na i na rozsah 0-100% bodů. Tento mód není vhodný pro generování Voronoiových diagramů z běžných obrázků (rozlišení okolo 1280x1024), proto je stisknutí tlačítka pro výpočet Voronoiovo diagramu v tomto módu je zobrazeno upozornění o tom, že výpočet může být delší.

Tento mód se ale velmi hodí pro samplování obrázků. Na rozsahu 0-1% máme po samplování jenom několik bodů, které nepůsobí na pohled nijak zvlášť hezky. Zatímco na rozsahu 0-100% samplování způsobuje docela zajímavé efekty.

3.2 Pravá část

Pravá část obsahuje dvě taby, jeden pro samplování a jeden pro Voronoiův diagram.

3.2.1 Tab sampleru



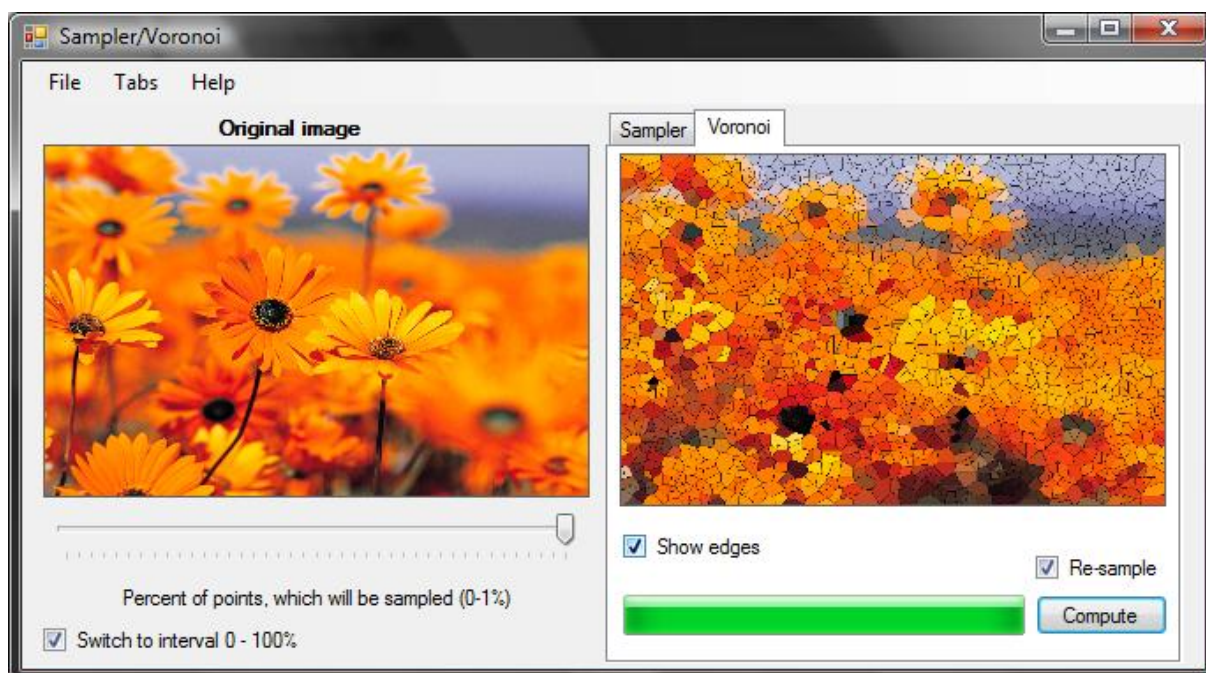
Na tomto tabu je možné provádět podrobnější nastavení sampleru. Je možné samplovat s rovnoměrným rozložením, s Gaussovským rozložením a po rovnoměrně rozložených clusterech, kde jednotlivé body clusteru mají Gaussovské rozložení.

Význam jednotlivých nastavení:

- Count – počet bodů, které cluster obsahuje
- Size – velikost clusteru
- Variance – rozptyl

Zaškrtnutí „Auto sample“ je možné použít pro automatické přesamplování v případě nějaké změny (např. změna počtu bodů, rozptyl, atp.)

3.2.2 Tab Voronoi



Na tomto tabu je možné z nasamplovaných dat vypočítat Voronoiův diagram. Tlačítkem Compute se spustí vlastní výpočet. Zaškrtnutí „Show edges“ umí zobrazit hrany Voronoiova diagramu.

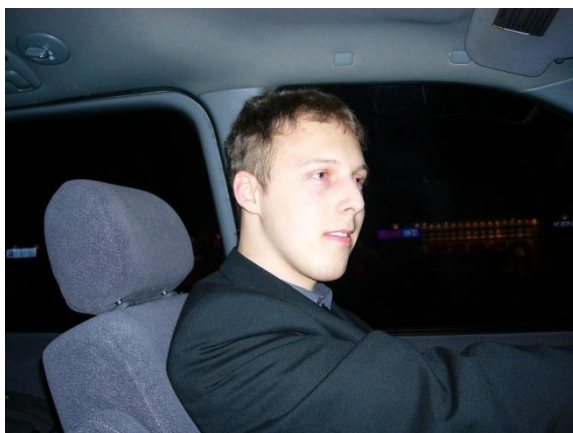
Zaškrtnutí „Re-sample“ je vhodné použít, pokud nechceme překlíkávat mezi do tabu Sampler pro vygenerování nové vstupní množiny. Pokud je toto zaškrtnutí zaškrtnuto, tak se automaticky provede přesamplování na základě aktuálně zadaných parametrů.

3.3 Klávesové zkratky

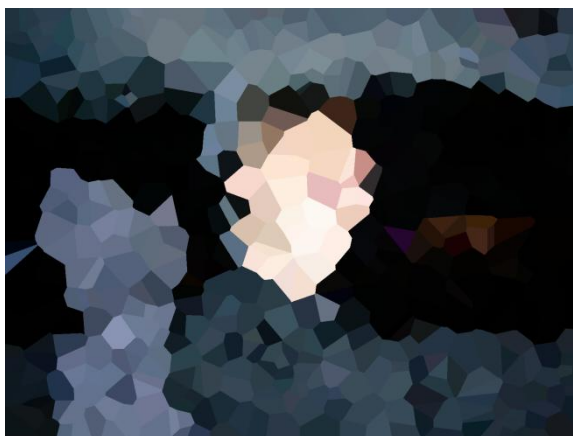
Aplikace klade důraz na klávesové zkratky, které zde vypíšu:

Enter	Nové samplování, popř. výpočet Voronoiova diagramu
Alt+1	Přepne na tab se samplerem
Alt+2	Přepne na tab s Voronoiem
Alt+x	Ukončení aplikace
Ctrl+o	Otevře vstupní bitmapu
Ctrl+s	Uloží výstup sampleru do bitmapy
Ctrl+Shift+s	Uloží výstup Voronoiova diagramu do bitmapy
Left Arrow	Posouvá posuvník pro počet bodů do leva
Right Arrow	Posouvá posuvník pro počet bodů do prava

4 Ukázka vstupního a výstupního obrázku



Originál



Voronoi

5 Použité zdroje

BenDi. Fortune's Voronoi algorithm implemented in C#. [Online]
<http://www.codeproject.com/KB/recipes/fortunevoronoi.aspx>.

Wikipedia. [Online] http://en.wikipedia.org/wiki/Fortune's_algorithm.