

# **Politechnika Warszawska**

Studia magisterskie

Optyczne techniki skanowania i analizy danych trójwymiarowych

DOKUMENTACJA

ETAP I

Zaznaczenie punktów znajdujących się na krawędziach  
chmury

inż. Jarosław Affek  
nr indeksu: 269259

Warszawa 2020

## 1. Wstęp

Celem projektu było opracowanie i zaimplementowanie w środowisku FRAMES algorytmu zaznaczającego punkty znajdujące się na wszystkich krawędziach chmury przy użyciu języka C++.

## 2. Algorytm

Po analizie kilku algorytmów zdecydowano się na algorytm opisany w artykule *Detecting Holes in Surfaces* autorstwa Ruwena Schnabela o nazwie *The weighted average criterion*.

Kroki algorytmu:

- określenie promienia sfery, wewnątrz której analizowane są punkty sąsiadujące z obecnie badanym punktem
- obliczenie odległości między punktami sąsiadującymi a punktem badanym
- dopasowanie płaszczyzny do badanych punktów wewnątrz sfery
- zrzutowanie wszystkich punktów na wyliczoną płaszczyznę
- obliczenie nowego punktu jako średniej ważonej sąsiedztwa (zrzuconego na płaszczyznę) badanego punktu, wg poniższych wzorów

$$g(d) = \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-d^2}{2\sigma^2}$$

gdzie:

$g$  – jądro Gaussa – jako waga poszczególnych punktów z sąsiedztwa

$d$  – odległość danego punktu sąsiadującego od punktu badanego (w przestrzeni 3D)

$\sigma$  – odległość między najdalszym punktem sąsiadującym a punktem badanym

$$w(q) = g(||p - prj(q)||)$$

gdzie:

$w$  – ważony składnik sumy

$g$  – waga (z wzoru powyżej)

$p$  – badany punkt

$q$  – punkt z sąsiedztwa

$prj(q)$  – punkt z sąsiedztwa zrzutowany na płaszczyznę

$||p - prj(q)||$  - odległość punktu  $p$  i  $q$  zrzutowanych na płaszczyznę

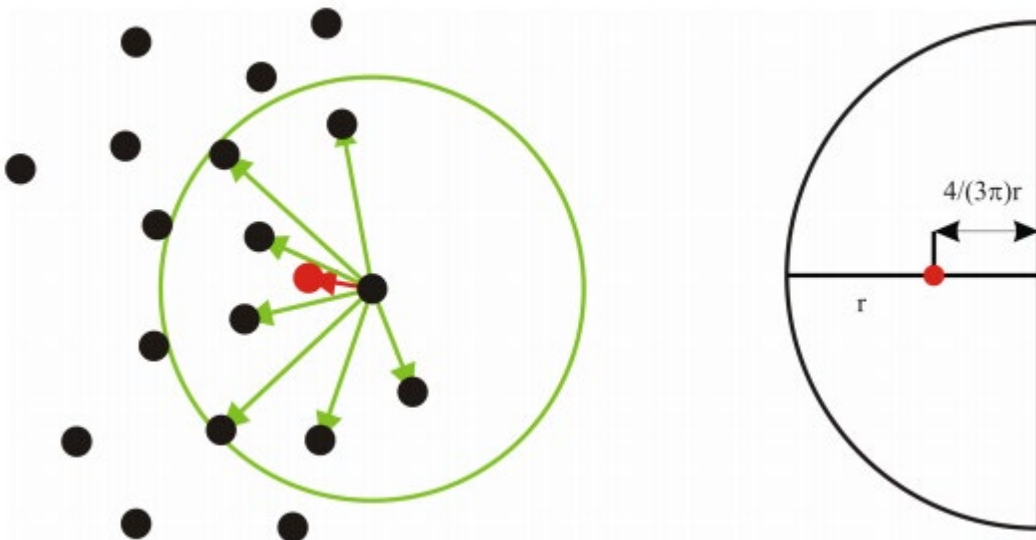
$$\mu = \frac{\sum_{i=1}^k w(q_i)q_i}{\sum_{i=1}^k w(q_i)}$$

gdzie:

$\mu$  – obliczony punkt

$w, q$  – symbole z wzorów powyżej

- porównanie położenia wyznaczonego punktu na płaszczyźnie do środka ciężkości półdysku, który powinien znajdować się w odległości  $\frac{4}{3\pi}r$  od punktu badanego (rysunek poniżej).



Rysunek 1

Porównanie następuje poprzez obliczenie „prawdopodobieństwa”, którego wartość zależy od ustawionego przez użytkownika progu określa czy punkt można określić punktem na krawędzi, Domyślna wartość prawdopodobieństwa określająca punkt jako krawędziowy to  $>0,9$ . Prawdopodobieństwo obliczane jest z poniższego wzoru:

$$P(p) = \min \left( \frac{\|p - prj(\mu)\|}{\frac{4}{3\pi}r}, 1 \right)$$

gdzie:

$P$  – prawdopodobieństwo

$\|p - prj(\mu)\|$  - odległość między wyliczonym punktem i badnym punktem zrzutowanymi na płaszczyznę

$$r = \frac{1}{2k} \sum_{i=1}^k \|p - prj(q)\|$$

gdzie:

$k$  – liczba sąsiadów

$\|p - prj(q)\|$  - odległość punktu badanego od punktu sąsiadującego zrzutowanych na płaszczyznę

### 3. Implementacja

Zgodnie z przedstawionym algorytmem zaimplementowano powyższe wzory w identycznej postaci jak opisano powyżej. Użytkownik ma możliwość wyboru promienia sfery, w ramach której analizowane są punkty sąsiadujące oraz próg prawdopodobieństwa, wg którego podejmowana jest decyzja o zaznaczeniu punktu jako krawędziowego. W programie tworzona jest warstwa danych *probability* zawierająca wartości prawdopodobieństwa dla każdego punktu. Ponadto w podobny sposób tworzona jest segmentacja. Użytkownik ma możliwość wyboru na ile grup mają być posegmentowane punkty. Segmentacja odbywa się

również na podstawie wartości prawdopodobieństwa każdego punktu i zapisywana do warstwy *segmentation*. Wybór liczby segmentów polega na podziale całego przedziału prawdopodobieństwa (0-1) na wybraną liczbę równych podprzedziałów.

#### **4. Podsumowanie**

Algorytm działa poprawnie, punkty krawędziowe są zaznaczane. Zależnie od wybranych przez użytkownika parametrów (zwłaszcza próg prawdopodobieństwa) można uzyskać większą lub mniejszą „czułość” i różną liczbę zaznaczonych punktów oraz odmienną „grubość” zaznaczonych krawędzi.

#### **5. Bibliografia**

- *Detecting Holes in Surfaces*, Ruwen Schnabel; Institute of Computer Graphics, University of Bonn, Germany - <https://old.cescg.org/CESCG-2005/papers/Bonn-Schnabel-Ruwen.pdf>