

# Vertex clustering

## Sprawozdanie

### 1. Problem upraszczania siatek

**Upraszczać siatek** można nazwać pewnym szczególnym rodzajem problemu tzw. remeshingu. Ogólnie, mając siatkę  $M = (V, F)$ , chcemy znaleźć siatkę  $M' = (V', F')$  taką, że:

- $|V'| < |V|$  oraz  $\|M - M'\|$  jest minimalna
- lub
- $\|M - M'\| < \varepsilon$  oraz  $|V'|$  jest minimalna.

Istnieje kilka podejść do tego problemu. Jednym z nich jest podejście oparte na klasteryzacji wierzchołków (ang. *vertex clustering*).

### 2. Algorytm

Zaimplementowany algorytm należy do grupy algorytmów jednokrokowych. Jest dość wydajny, ale jakość siatek wyjściowych często nie jest zbyt satysfakcyjną - często trzeba odpowiednio dobierać parametry algorytmu. Pseudokod algorytmu wygląda następująco:

1. Niech będzie dana siatka wyjściowa  $M = (V, F)$ .
2. Podziel przestrzeń otaczającą obiekt na komórki (klastry; w kodzie nazywamy je *buckets*) - sześciany o boku  $2\varepsilon$ .
3. Dla każdego wierzchołka  $v \in V$  siatki oryginalnej ( $M$ ):
  - a. Wyznacz klaster  $b$ , do którego należy  $v$ .
  - b. Dodaj  $v$  do  $b$ .
4.  $V' = \emptyset$
5. Dla każdego klastra  $b$ :
  - a. Wyznacz reprezentanta  $r_b$  klastra.
  - b.  $V' = V' \cup \{r_b\}$
6.  $F' = \emptyset$
7. Dla każdego elementu  $f \in F$  siatki oryginalnej ( $M$ ):
  - a.  $B_f = \emptyset$
  - b. Dla każdego wierzchołka  $v \in f$ :
    - i. Wyznacz klaster  $b$ , do którego należy  $v$ .
    - ii.  $B_f = B_f \cup \{b\}$
  - c. Jeśli  $B_f$  jest trójkątem:
    - i.  $F' = F' \cup \{B_f\}$
8. Siatka wynikowa to  $M' = (V', F')$ .

Jak można zauważyc, algorytm ma dwa parametry - wartość stałej  $\epsilon$  oraz metodę wyznaczania reprezentanta klastra.

Wartość  $\epsilon$  określa wielkość klastrów, ale należy ją odpowiednio dobrac (każdy mesh ma różną "rozpiętość" - zatem dla każdej siatki  $\epsilon$  będzie zależał od tego, jakie są odległości między jej wierzchołkami).

Zaimplementowaliśmy 4 metody wyznaczania reprezentanta:

- **"Dummy" method** - jako reprezentanta przyjmij środek klastra.
- **Mean** - jako reprezentanta przyjmij punkt o współrzędnych  $(x_{mean}, y_{mean}, z_{mean})$ , gdzie  $a_{mean}$  oznacza średnią wartość współrzędnej  $a$ ,
- **Median** - jako reprezentanta przyjmij wierzchołka z klastra, który leży najbliżej punktu  $(x_{median}, y_{median}, z_{median})$ , gdzie  $a_{median}$  oznacza medianę współrzędnej  $a$ ,
- **Quadric errors** - jako reprezentanta przyjmij punkt o takich współrzędnych, by zminimalizować sumę odległości do płaszczyzn zawierających sąsiednie trójkąty wszystkich wierzchołków z klastra.

Należy też zaznaczyć, że nasz algorytm działa **tylko dla siatek trójkątnych**, gdyż jest to bardzo duże ułatwienie przy wyliczaniu reprezentanta oraz wyznaczaniu elementów siatki wynikowej.

## 2.1. Metoda quadric errors

Jest to dość kosztowna metoda, ale pozwala na poprawne zachowanie się algorytmu przy ostrzych krawędziach modelu.

Polega ona na minimalizacji tzw. błędu średniokwadratowego. Dla i-tego sąsiedniego elementu wierzchołków należących do klastra obliczamy wektor normalny  $n_i$  - jednostkowy wektor prostopadły do płaszczyzny. Można to zrobić np. przez zastosowanie iloczynu wektorowego wektorów  $v_1 v_2$  oraz  $v_1 v_3$  i normalizację wektora wynikowego. Oznaczając  $d_i = n_i^T u_1$ , gdzie  $u_1$  to wektor o współrzędnych punktu  $v_1$ , możemy obliczyć kwadrat odległości punktu  $x$  od płaszczyzny zawierającej i-ty sąsiedni element, korzystając ze wzoru:  $dist^2(x, f_i) = (n_i^T \cdot x - d_i)^2$ . Po przekształceniach, można też otrzymać układ równań:

$$\underbrace{\left( \sum_i \mathbf{n}_i \mathbf{n}_i^T \right)}_{\mathbf{A}} \mathbf{x} = \underbrace{\left( \sum_i \mathbf{n}_i d_i \right)}_{\mathbf{b}}$$

Jeśli weźniemy rozwiązanie średniokwadratowe powyższego układu, otrzymamy reprezentanta klastra  $b$ .

Rozwiązanie takie można otrzymać, obliczając tzw. pseudoinwersję  $A^+$  macierzy  $A$ . Wówczas reprezentanta można obliczyć jako  $r_b = A^+ \cdot b$ .

Poniżej przedstawiamy pseudokod tej metody:

1. Dane:  $b$  - kластер, dla którego szukamy reprezentanta.
2. Niech  $A$  będzie macierzą zerową  $3 \times 3$  oraz  $b$  - wektorem zerowym w 3D.
3. Niech  $F_b = \{f \in F : \exists v \in b : v \in f\}$  - zbiór sąsiadów wierzchołków z  $b$ .
4. Dla każdego elementu  $f = (v_1, v_2, v_3) \in F_b$ :
  - a.  $n_f = v_1 v_2 \times v_1 v_3$ .
  - b.  $n_f = \frac{n_f}{\|n_f\|}$
  - c.  $d_f = n_f^T \cdot u_1$ , gdzie  $u_1$  to wektor o współrzędnych punktu  $v_1$
  - d.  $A = A + n_f \cdot n_f^T$
  - e.  $b = b + d_f \cdot n_f$
5. Reprezentant bucketu  $b$ :  $r_b = A^+ \cdot b$ , gdzie  $A^+$  to tzw. pseudoinwersja macierzy  $A$  - taka macierz, która "rozwiązuje" problem błędu średniokwadratowego dla równania  $Ax = b$ .

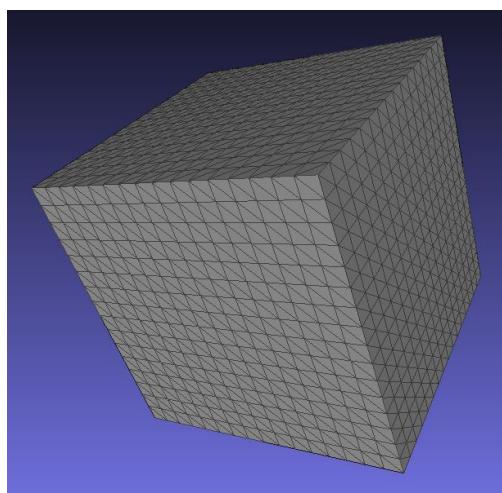
### 3. Użyte struktury danych

Algorytm został zaimplementowany w języku Python. W projekcie wykorzystana została struktura Half-edge w implementacji z biblioteki CGAL. Do działań na wektorach i macierzach użyliśmy biblioteki numpy i jej struktury *ndarray*. Program umożliwia wczytanie siatek zapisanych w formatach OBJ oraz OFF.

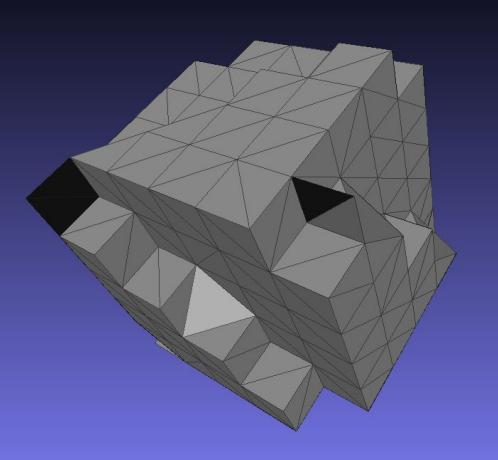
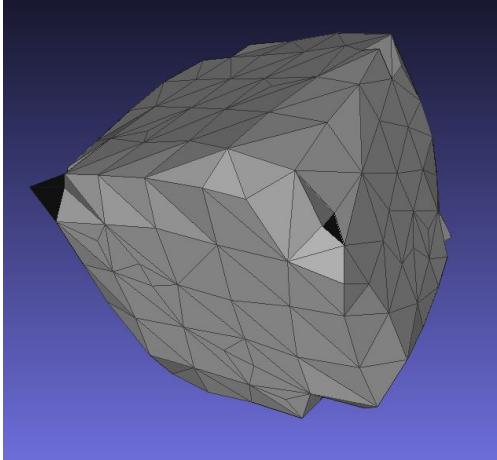
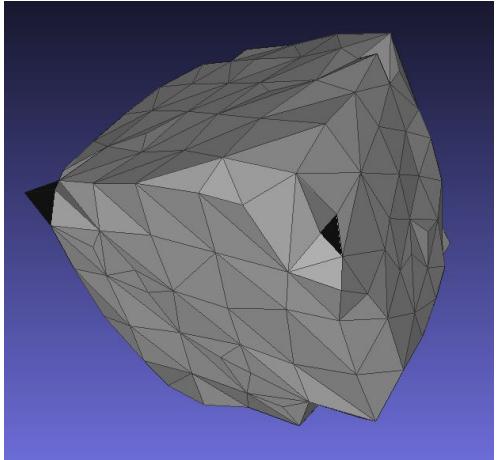
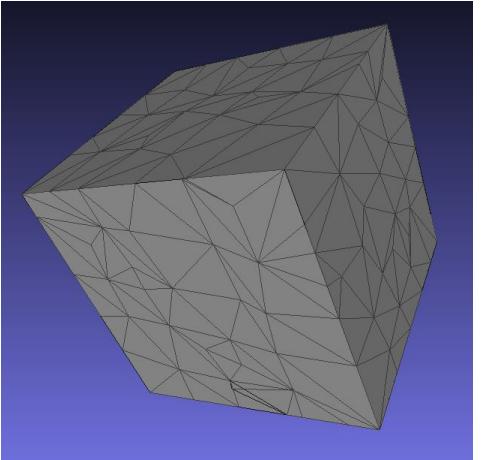
### 4. Siatki testowe

#### 4.1. Sześcian

Siatka reprezentuje sześciyan składający się z 2116 wierzchołków i 3072 trójkątów. Sześciyan jest lekko obrócony względem wszystkich trzech osi. Długość boku wynosi ok. 10 jednostek. W testach przyjęliśmy epsilon = 2.0.



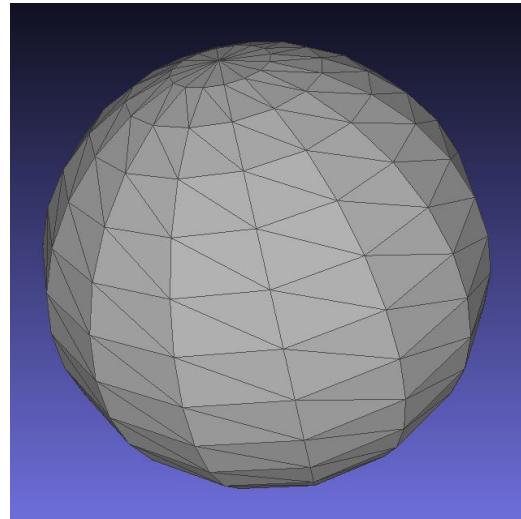
Oryginalna siatka

	
Cluster center	Mean
	
Median	Quadric errors

Zdecydowanie najlepszy rezultat dała metoda "Quadric errors" - zachowane zostały krawędzie bryły oraz ściany. Metoda "Cluster center" ma częściowo proste ściany, jednak są one obrócone względem oryginalnych. Wynika to z obrócenia oryginalnej siatki względem siatki klasteryzacji.

## 4.2. Sfera

Siatka reprezentuje sferę składającą się z 242 wierzchołków i 480 trójkątów. Średnica wynosi ok. 10 jednostek. W testach przyjęliśmy epsilon = 2.0. Wynikowe siatki, z wyjątkiem uzyskanej metodę "Cluster center" są bardzo zbliżone.

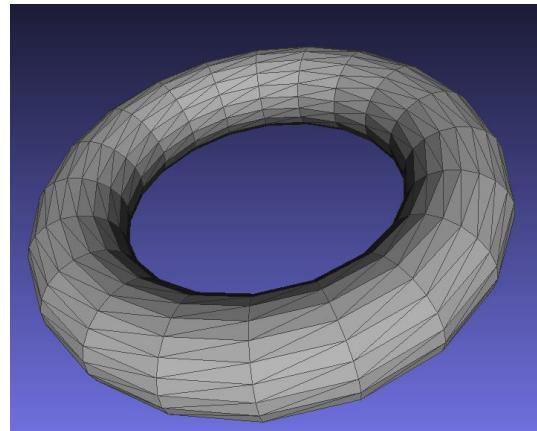


Oryginalna siatka

A grayscale polygonal sphere showing significant distortion, appearing elongated and irregular, centered against a blue gradient background.	A grayscale polygonal sphere with a more uniform distribution of vertices compared to the cluster center, centered against a blue gradient background.
Cluster center	Mean
A grayscale polygonal sphere that appears slightly more rounded than the mean, centered against a blue gradient background.	A grayscale polygonal sphere that appears slightly more rounded than the mean, centered against a blue gradient background.
Median	Quadric errors

### 4.3. Torus

Siatka reprezentuje torus składający się z 400 wierzchołków i 800 trójkątów. Średnica wynosi ok. 10 jednostek. W testach przyjęliśmy epsilon = 1.0. Wyniki są porównywalne z osiągniętymi dla siatki sferycznej.



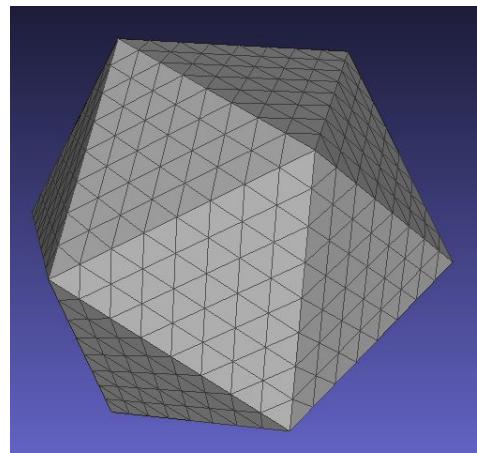
Oryginalna siatka

A 3D rendering of the torus mesh where the vertices have been moved towards a single central point, resulting in a highly distorted and non-manifold shape.	A 3D rendering of the torus mesh where the vertices have been moved towards the mean position of their neighbors, resulting in a smoother but still somewhat noisy surface.
Cluster center	Mean
A 3D rendering of the torus mesh where the vertices have been moved towards the median position of their neighbors, resulting in a smoother surface with fewer artifacts than the mean method.	A 3D rendering of the torus mesh where the vertices have been moved towards the minimum of a quadric error function, resulting in a very smooth and visually appealing surface.
Median	Quadric errors

#### 4.4. Dwudziestościan

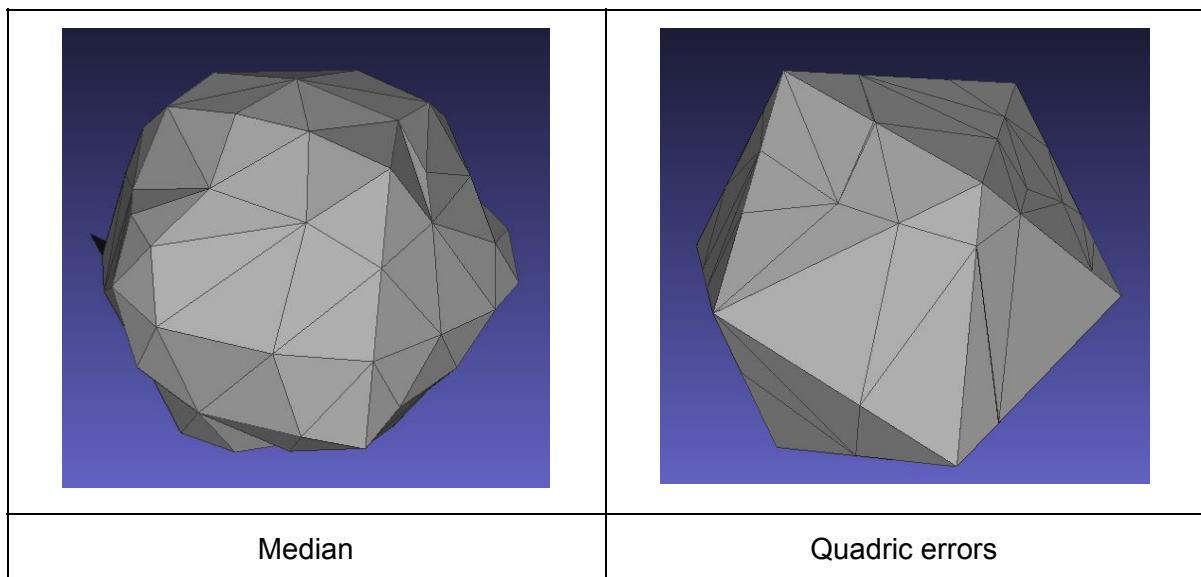
Siatka reprezentuje dwudziestościan składający się z 817 wierzchołków i 1280 trójkątów. Średnica wynosi ok. 10 jednostek. W testach przyjęliśmy epsilon = 2.0. Uproszczone siatki posiadają 85 wierzchołków i 170 trójkątów.

Jedyną metodą, która daje prawidłowy rezultat jest "Quadric errors". Metody średnich i median dały na wyjściu siatkę bardziej zbliżoną do kuli. Utracone zostały wszystkie krawędzie bryły oraz oryginalne ściany.



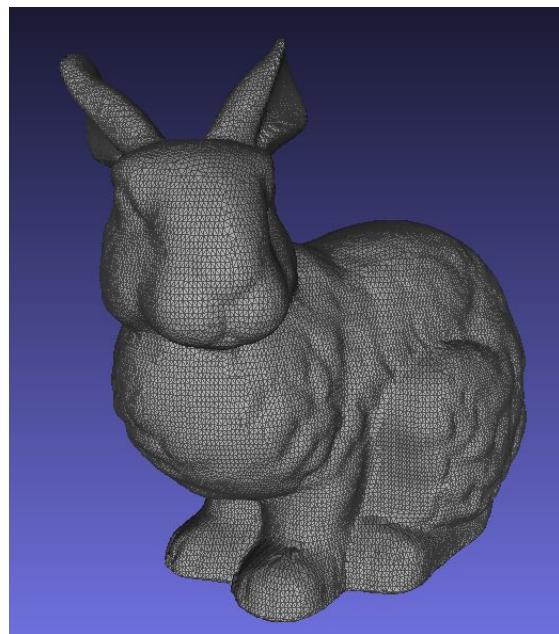
Oryginalna siatka

A 3D rendering of a dodecahedron mesh that has been significantly simplified. It appears more rounded and spherical than the original, with fewer facets and vertices. This represents the result of a clustering algorithm where the data points have been grouped into a single central cluster.	A 3D rendering of a dodecahedron mesh that has been smoothed or averaged. The result is a much smoother surface with many more facets and vertices than the original, giving it a more spherical appearance. This represents the result of a mean-based averaging process.
Cluster center	Mean

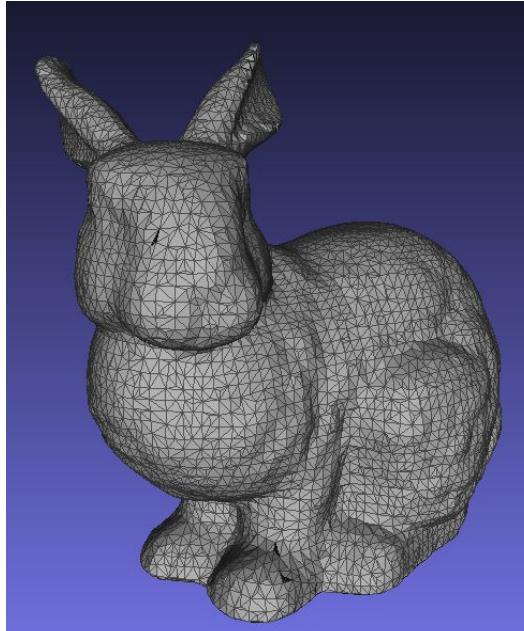
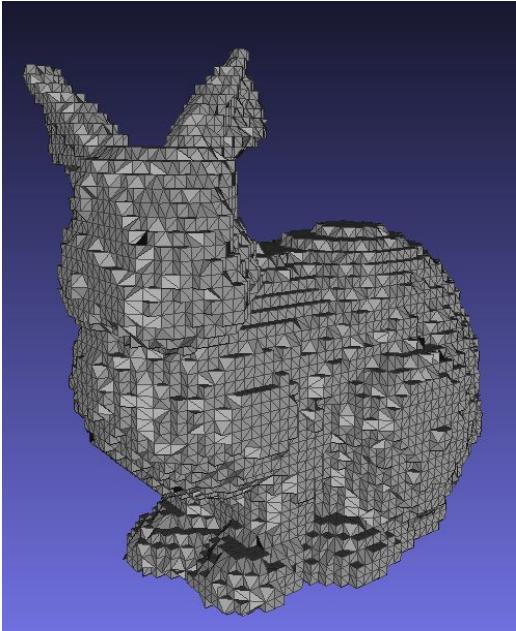


#### 4.5. Królik

Oryginalna siatka składa się z 35947 wierzchołków i 69451 trójkątów. Testy przeprowadzono dla wartości epsilon = 0.003. Wynikowe siatki składają się z 7798 wierzchołków i 15589 trójkątów.

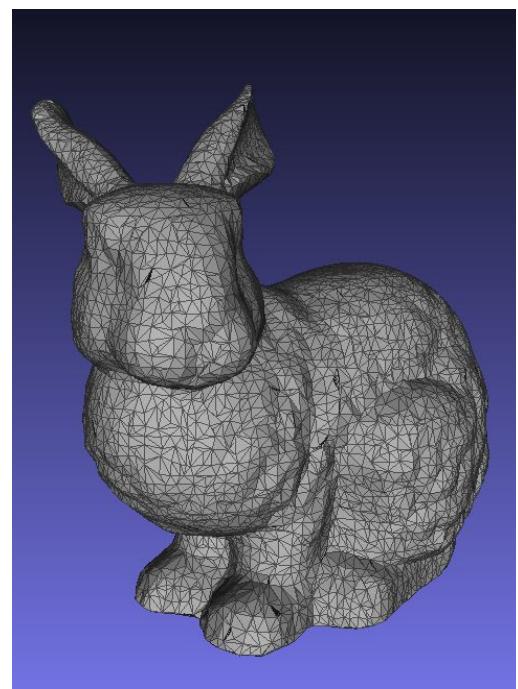
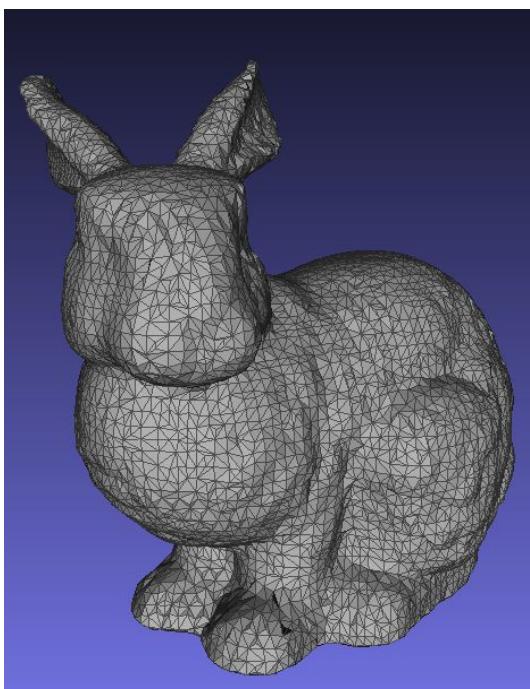


Oryginalna siatka



Cluster center

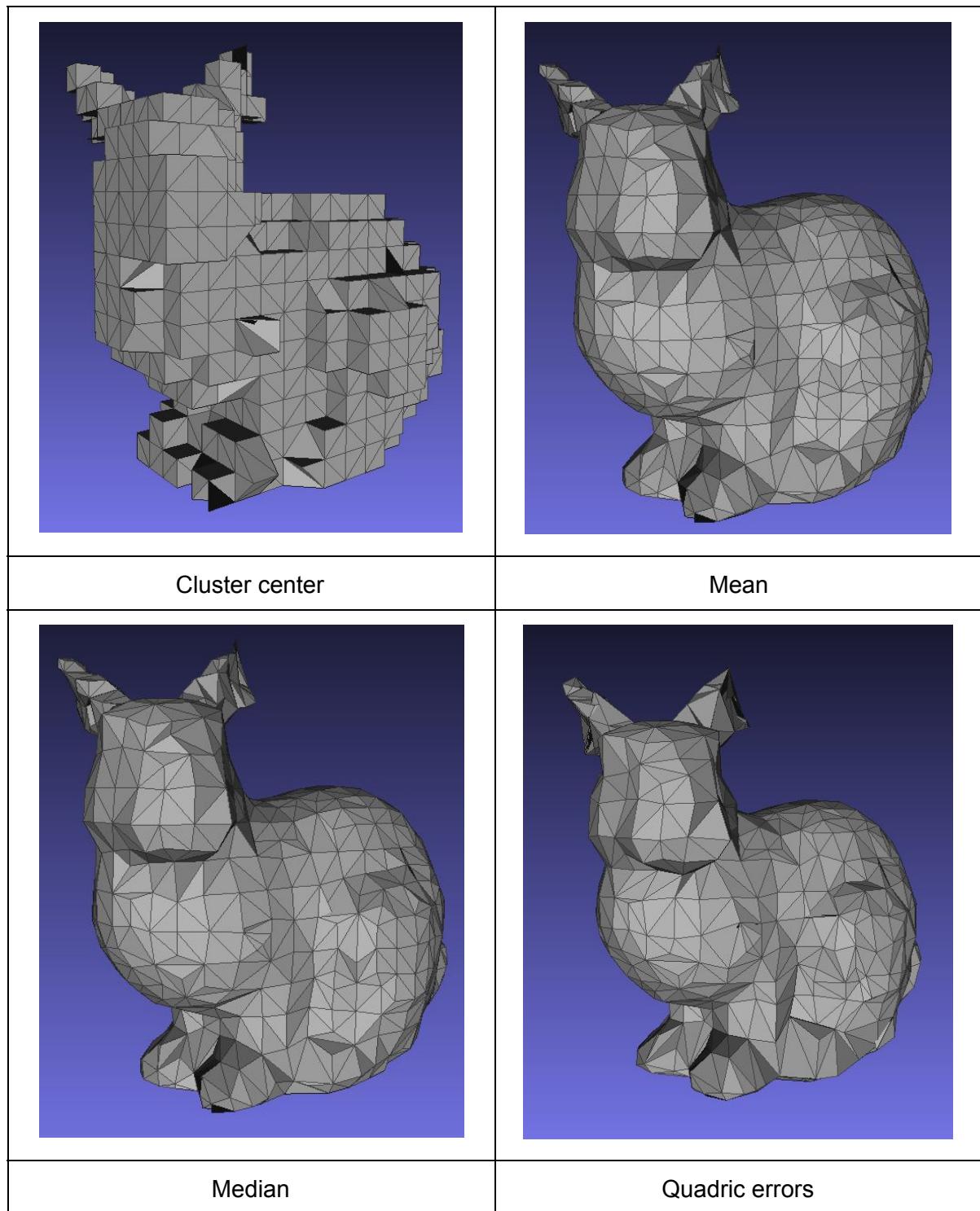
Mean



Median

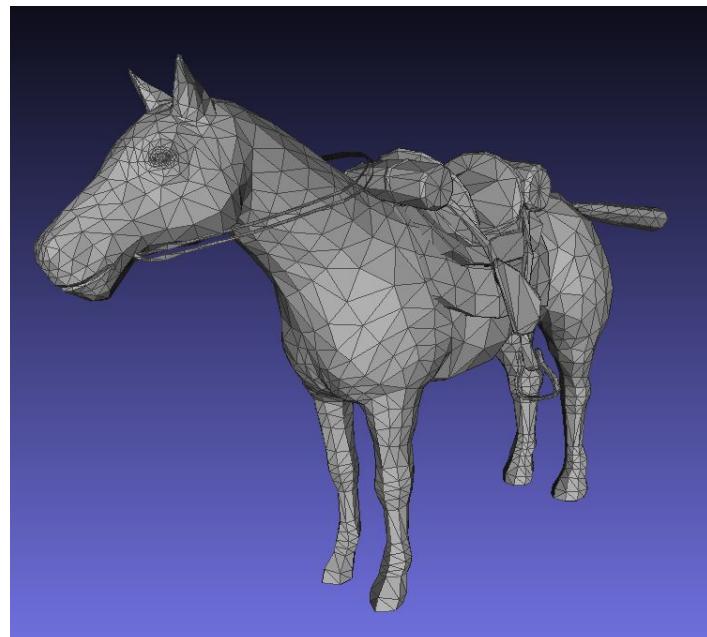
Quadric errors

Epsilon = 0.01, 799 wierzchołków, 1630 trójkątów:



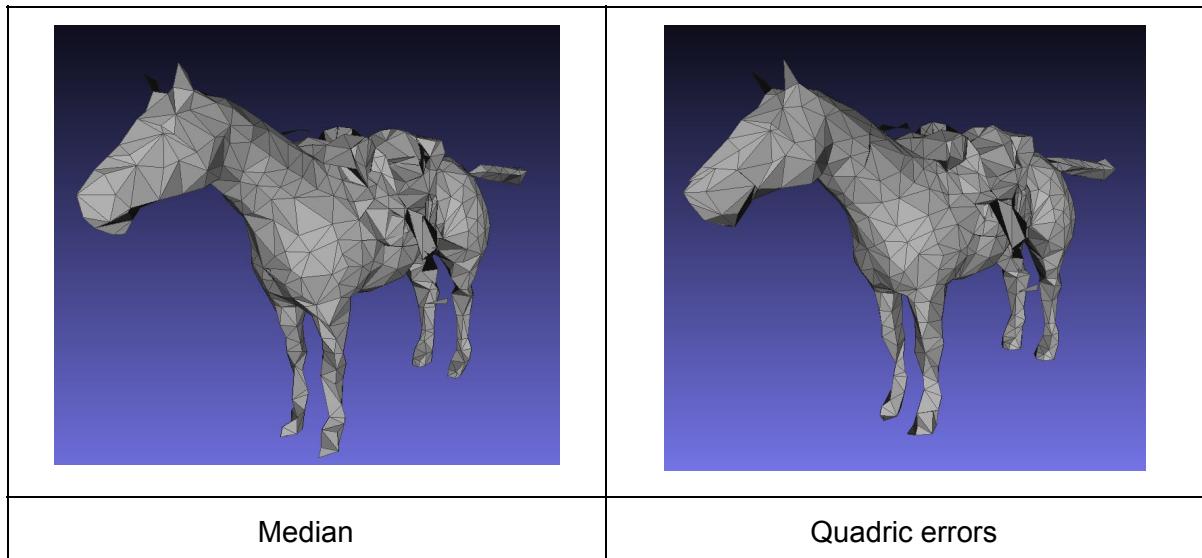
## 4.6. Koń

Oryginalna siatka składa się z 2831 wierzchołków i 5333 trójkątów. Testy przeprowadzono dla wartości epsilon = 10.0. Wynikowe siatki składają się z 886 wierzchołków i 1922 trójkątów.

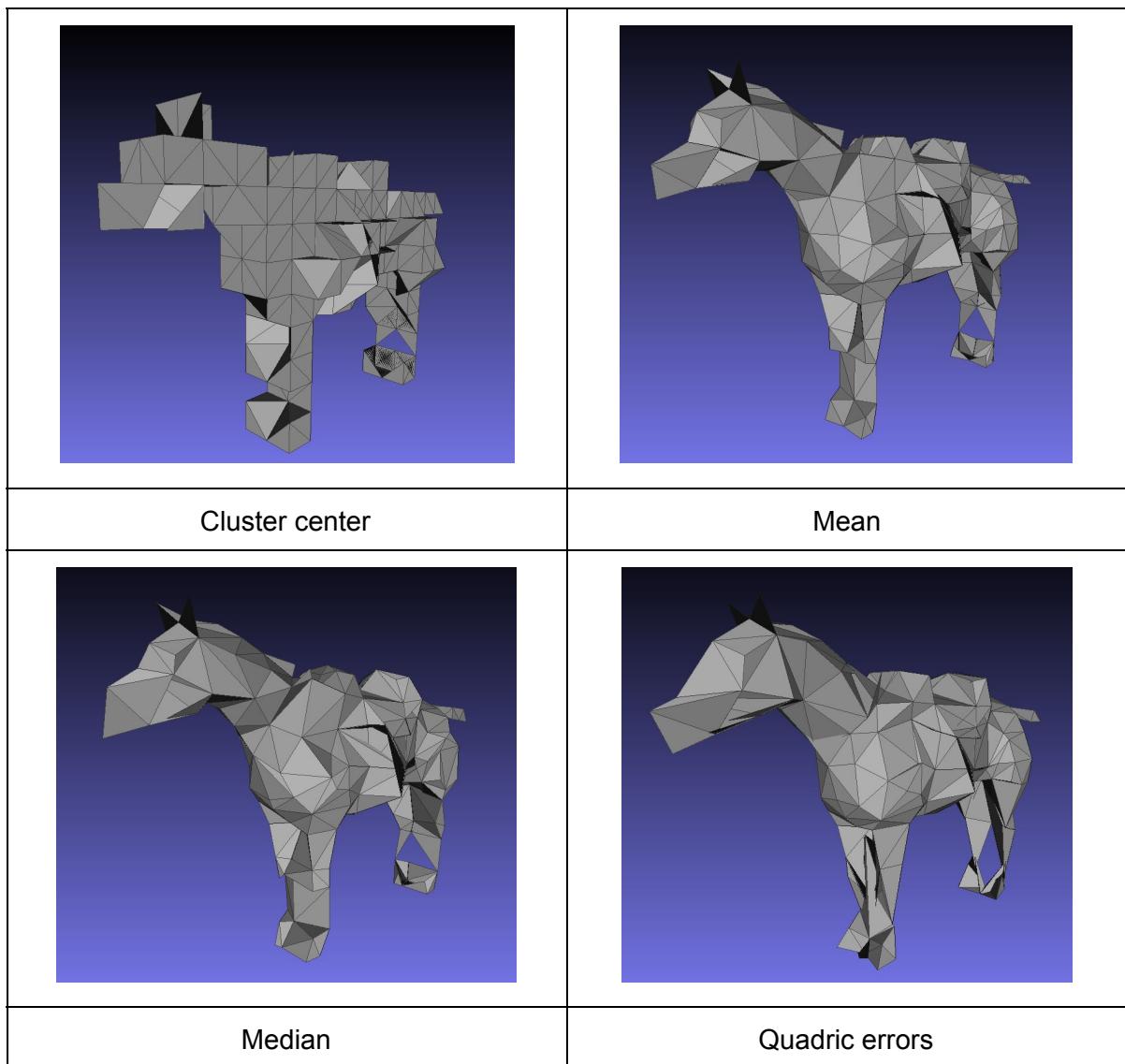


Oryginalna siatka

A 3D wireframe model of a horse standing in profile, facing left. The mesh is highly compressed and distorted compared to the original, appearing blocky and fragmented.	A 3D wireframe model of a horse standing in profile, facing left. The mesh is moderately compressed and distorted, appearing more continuous than the cluster center but less detailed than the original.
Cluster center	Mean

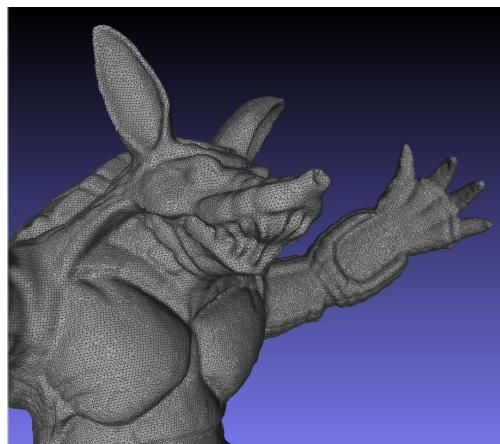


Epsilon = 20.0, 257 wierzchołków, 646 trójkątów:



## 4.7. Armadillo

Oryginalna siatka składa się z 172974 wierzchołków i 345944 trójkątów. Testy przeprowadzono dla wartości epsilon = 2.0. Wynikowe siatki składają się z 12774 wierzchołków i 25979 trójkątów.



Oryginalna siatka

A low-resolution wireframe version of the armadillo model, where the mesh has been significantly simplified, resulting in a blocky and less detailed appearance.	A wireframe model where the mesh has been averaged, creating a smoother surface compared to the cluster center but still maintaining some angular artifacts.
Cluster center	Mean
A wireframe model where the mesh has been averaged, resulting in a very smooth and rounded surface, though some original detail is lost.	A wireframe model where the mesh has been optimized to minimize quadric errors, resulting in a more balanced and visually appealing mesh than the mean or cluster center versions.
Median	Quadric errors

## 5. Testy czasowe

Test	Liczba wierzchołków	Średni czas wykonania [s]			
		Cluster center	Mean	Median	Quadric errors
<b>cube</b>	<b>2 116</b>	0.150	0.175	0.183	0.715
<b>sphere</b>	<b>242</b>	0.024	0.029	0.030	0.134
<b>bunny</b>	<b>35 947</b>	3.144	3.439	3.568	13.872
<b>horse</b>	<b>2 831</b>	0.251	0.287	0.311	1.231
<b>armadillo</b>	<b>172 974</b>	16.334	18.611	19.711	77.557

## 6. Wnioski

- Zaimplementowany algorytm ma dobrą złożoność obliczeniową:  $O(|V| \cdot O(repr_f) + |F|)$
- Dla prostych metod wyznaczania reprezentantów (*cluster center, mean, median*) złożoność ta jest równa w średnim przypadku  $O(|V| + |F|)$ .
- Dla metody *quadric errors* jest ona nieco gorsza - może wynieść  $O(|V| \cdot |F|)$ .
- Algorytm działa poprawnie, jednak dobór parametrów jest kluczowy i mocno zależy od siatki wyjściowej.
  - Epsilon trzeba dobrać w zależności od rozpiętości siatki i żądanej liczby wierzchołków.
  - Metodę wyznaczania reprezentanta należy dobrać w zależności od geometrii modelu.
- Oczywiście, metoda nie jest uniwersalna, np. kiepsko radzi sobie z modelami, w których są dziury.
- W przypadku siatek organicznych równie dobre wyniki dają wszystkie metody z wyjątkiem *cluster center*. Siatki posiadające wyraźne, ostre krawędzie oraz płaskie ściany wymagają zastosowania metody *quadric errors*, która pozwala je zachować.
- Dobór epsilon steruje ilością wierzchołków (i trójkątów) w wyjściowej siatce, natomiast dobór metody wyznaczania reprezentanta steruje zachowaniem własności geometrycznych siatki wynikowej.