

# **Tigri**

## Segmentare imagini pe baza texturilor

[https://github.com/madalinporojan2010/tiger\\_detection\\_opencv](https://github.com/madalinporojan2010/tiger_detection_opencv)

Studenti:

Porojan Mădălin

Pupeză Bogdan

Todor Andrei

Voivod Alin

În algoritmul nostru am urmărit pașii de mai jos:

- Se calculează patch-urile:

  - se calculează feature-urile patch-ului folosind GLCM;

  - se calculează histograma patch-ului;

- Se apelează kMeansClustering:

  - se calculează cele k clustere cu ajutorul funcției de euristică (Euclidian Distance/Cosine Similarity);

  - calcularea centroizilor se repetă de un număr de iterații (introdus de utilizator);

- La final se afișează imaginea rezultat.

## EXPERIMENTE

Test #1:

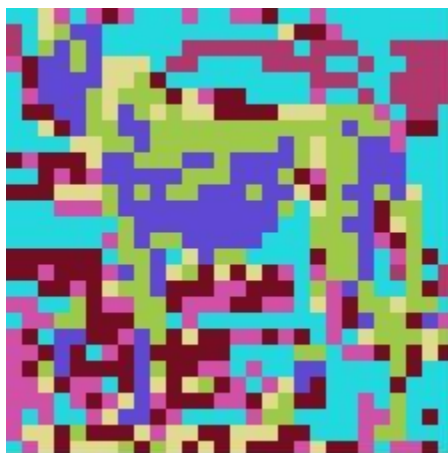
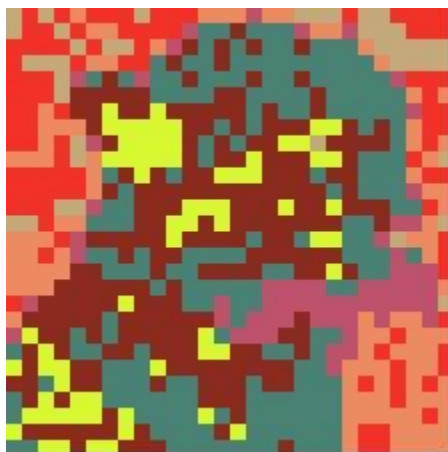
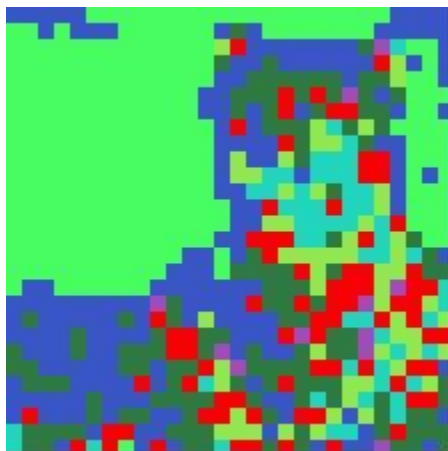
Iterații: 52

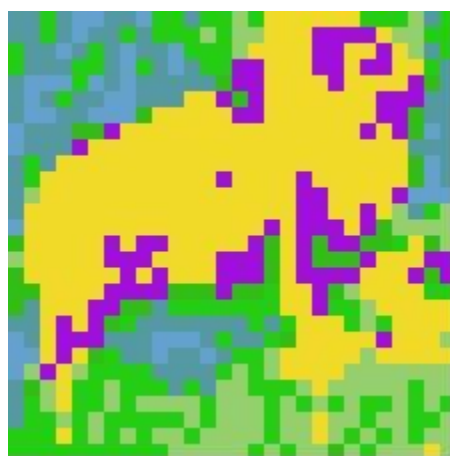
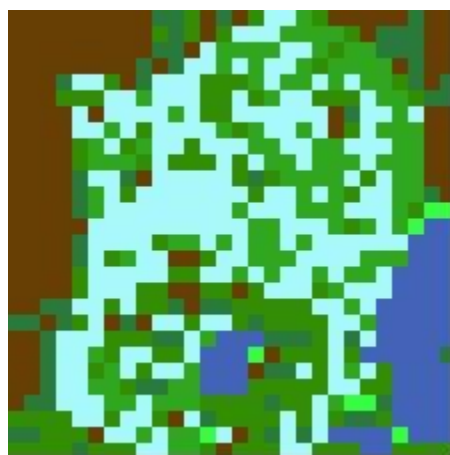
Patch size: 8

Clustere: 7

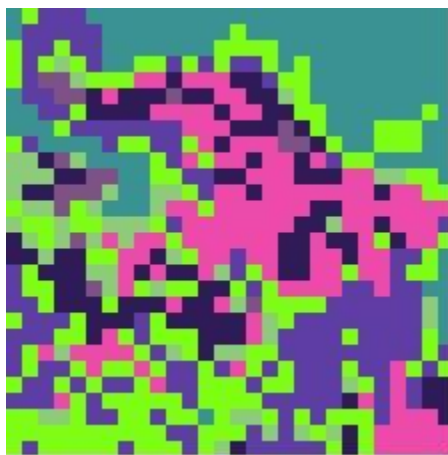
Heuristic function: Cosine Similarity Distance

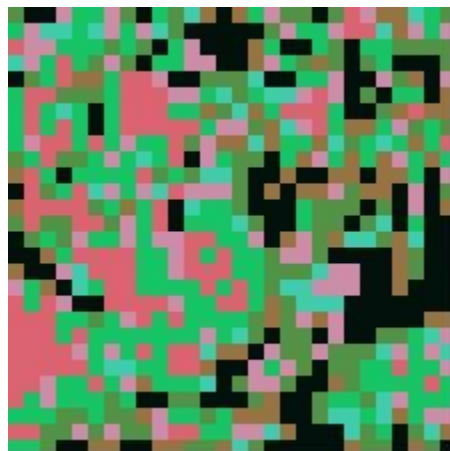












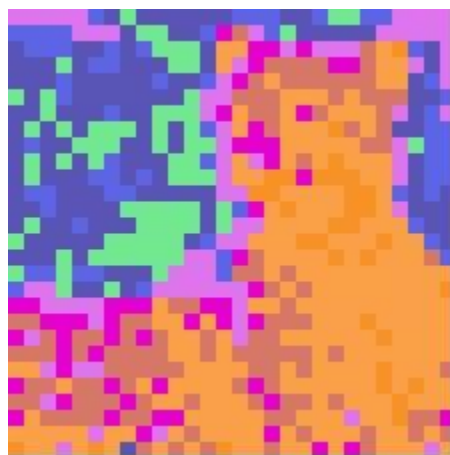
Test #2:

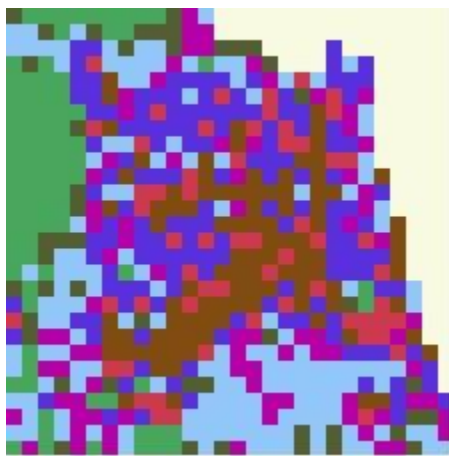
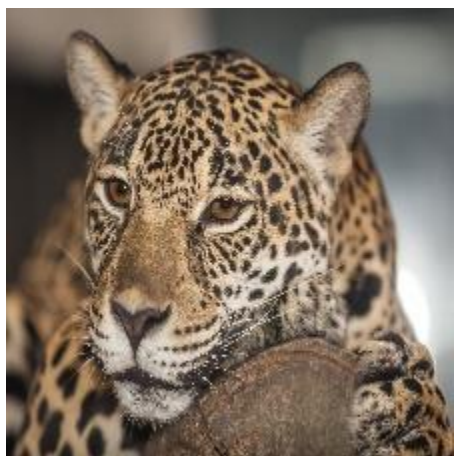
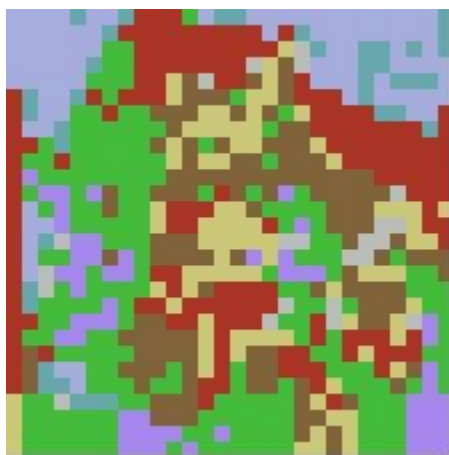
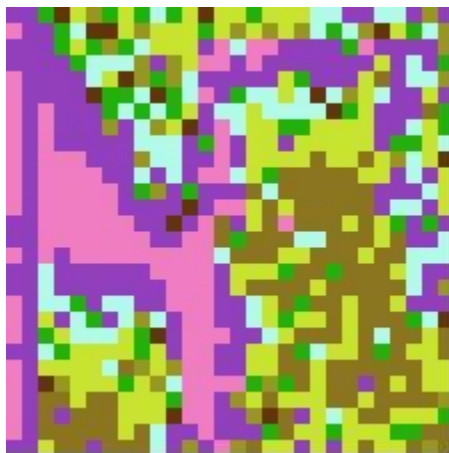
Iterații: 35

Patch size: 8

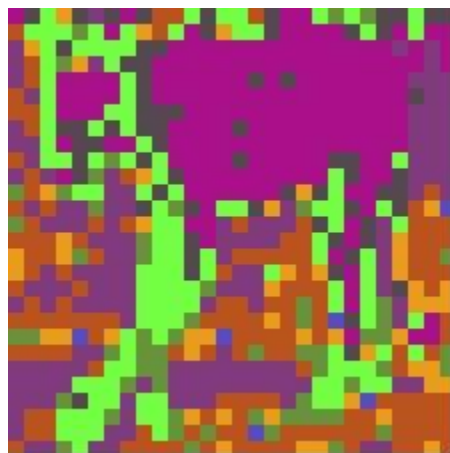
Clustere: 8

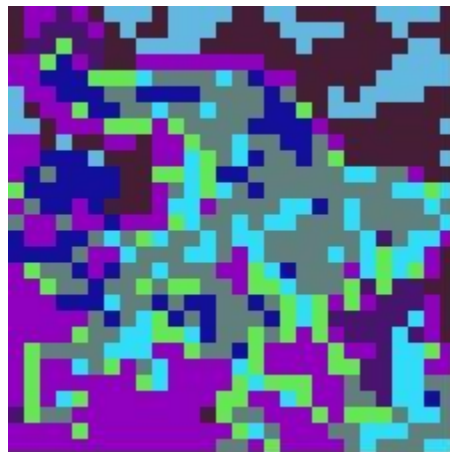
Heuristic function: Cosine similarity distance











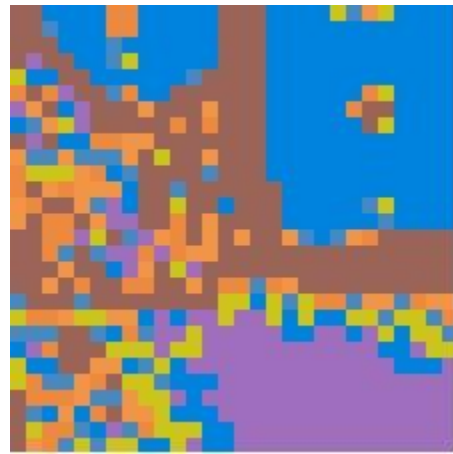
Test #3:

Iterations: 26

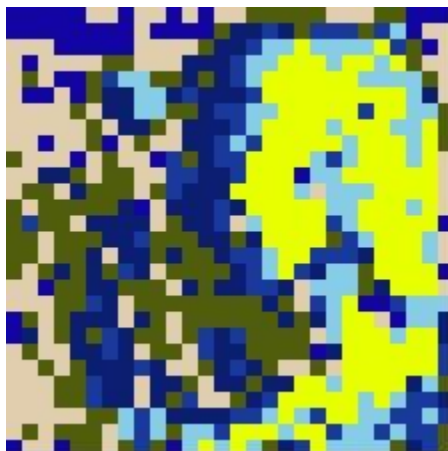
Patch size: 8

Number of clusters: 7

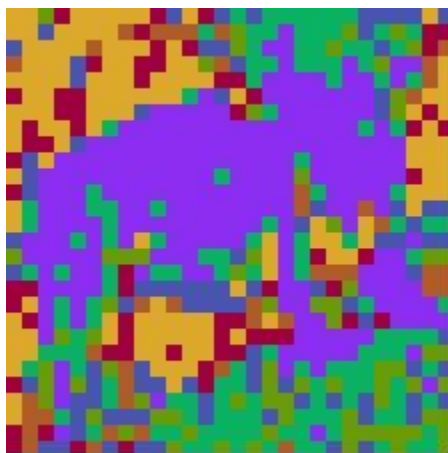
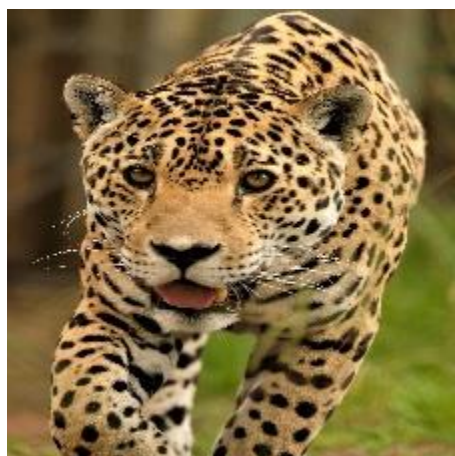
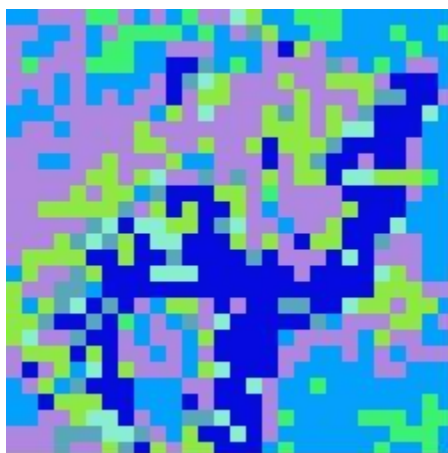
Heuristic function used: euclidian distance

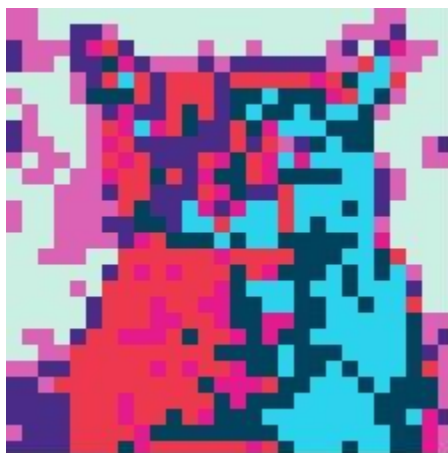
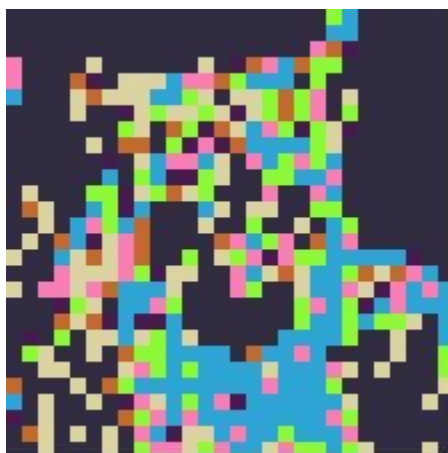
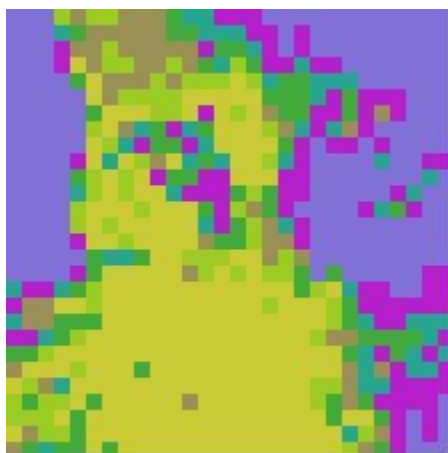














Test #4:

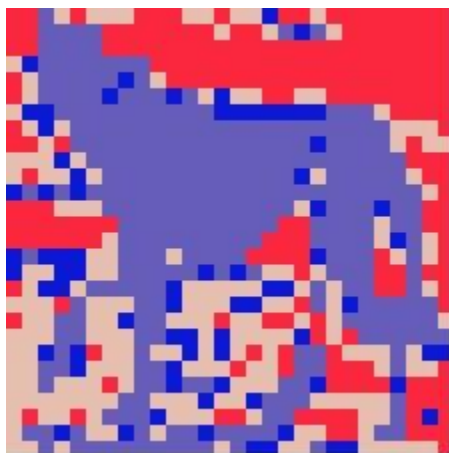
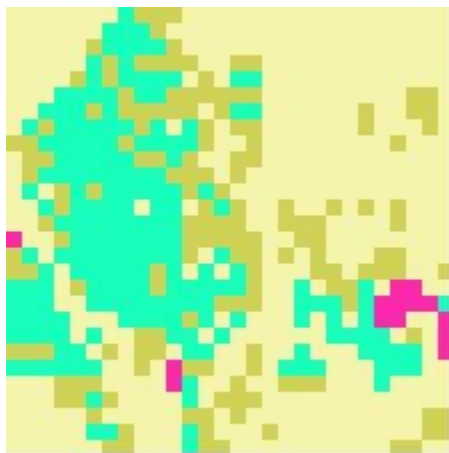
Iterations: 20

Patch size: 8

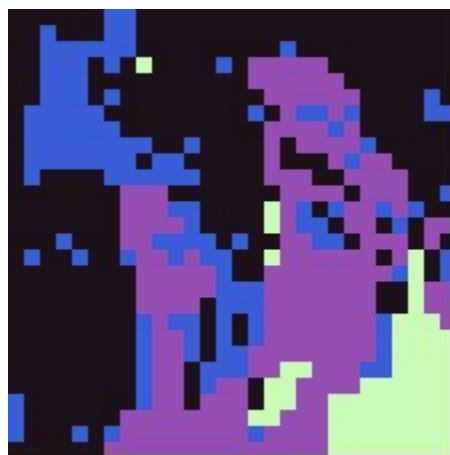
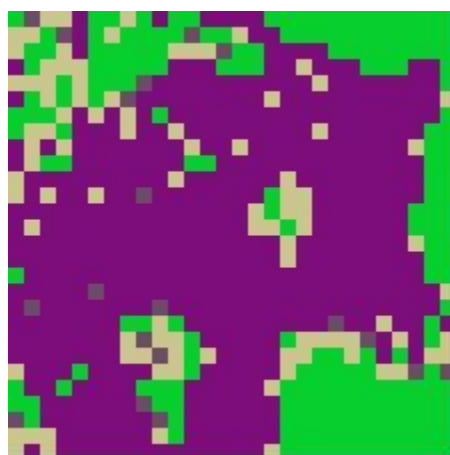
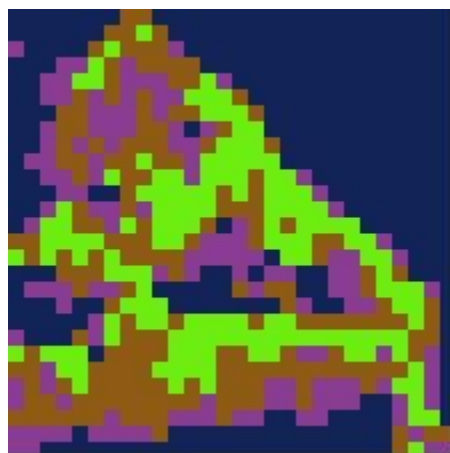
Number of clusters: 4

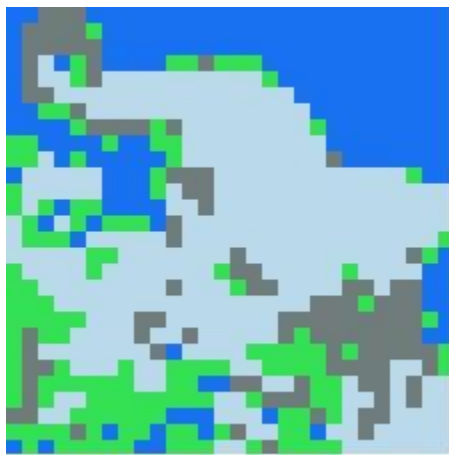
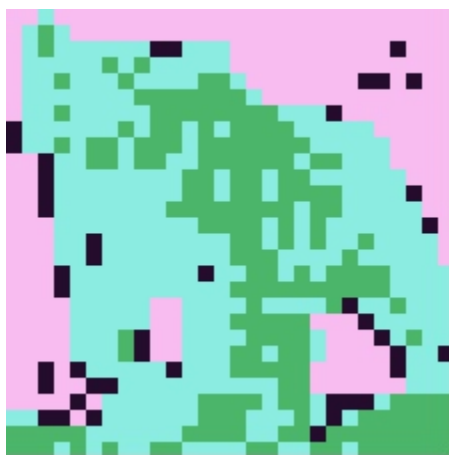
Heuristic function used: cosine similarity distance

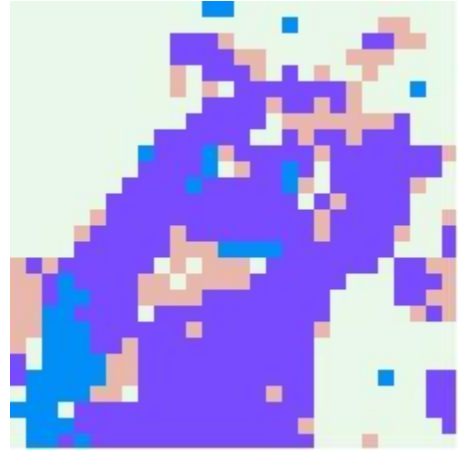












Test #5:

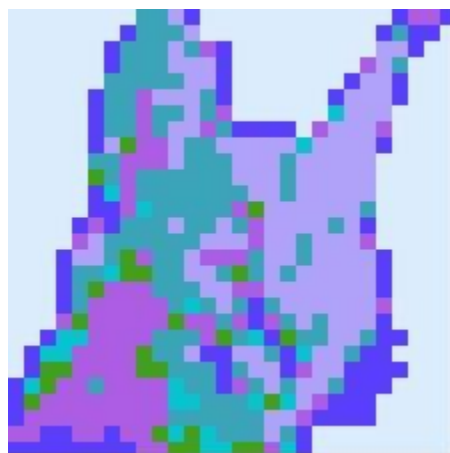
Iterations: 34

Patch size: 8

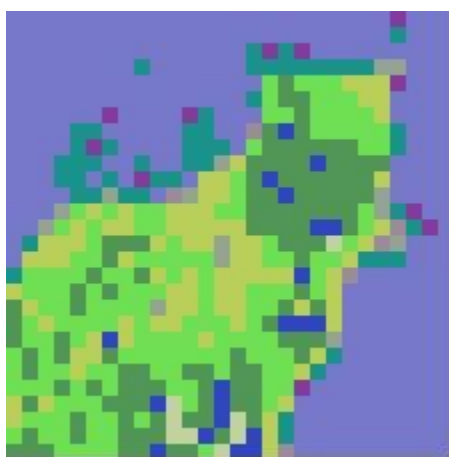
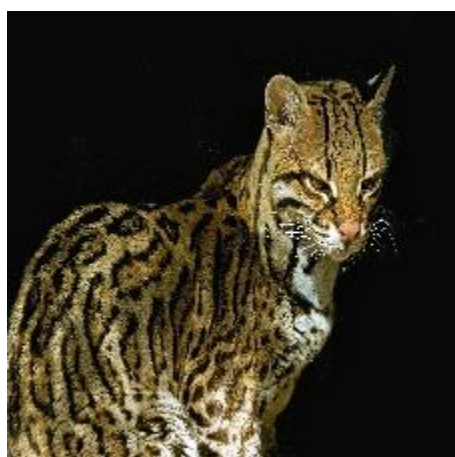
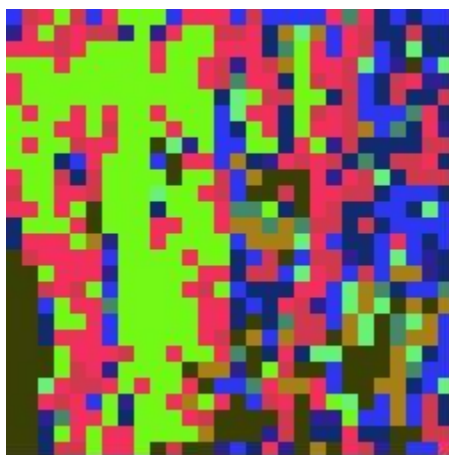
Number of clusters: 10

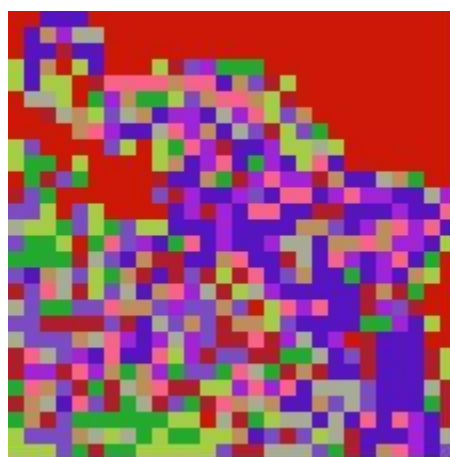
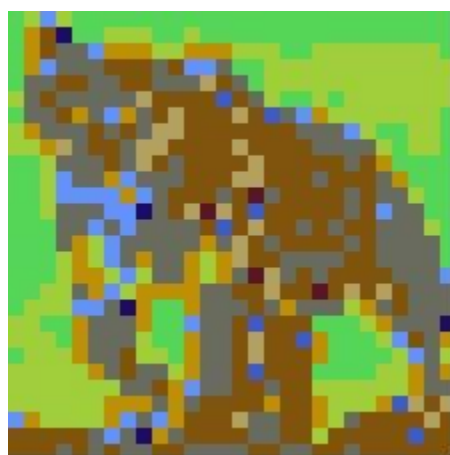
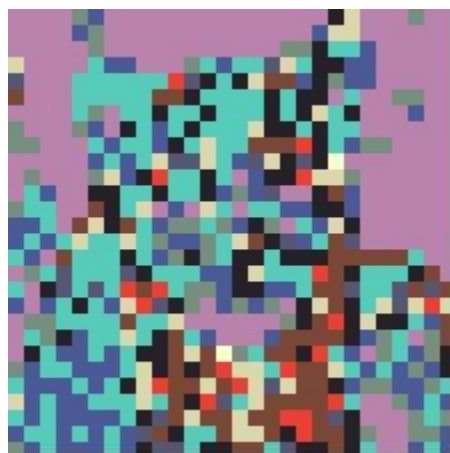
Heuristic function used: euclidian distance

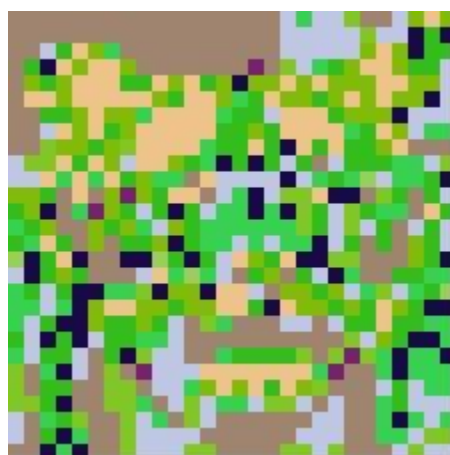














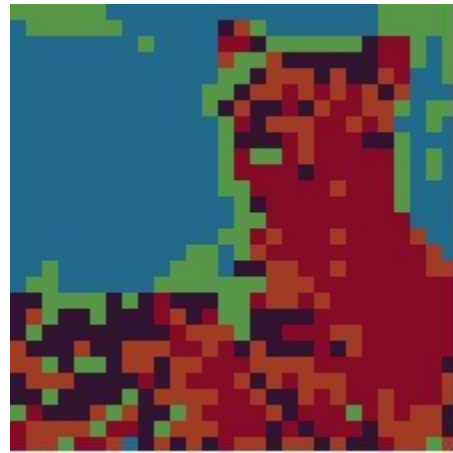
Test #6:

Iterations: 22

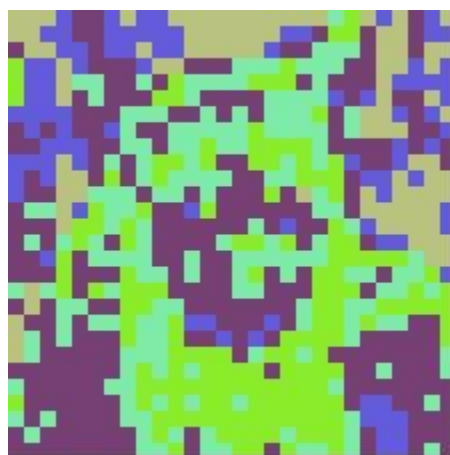
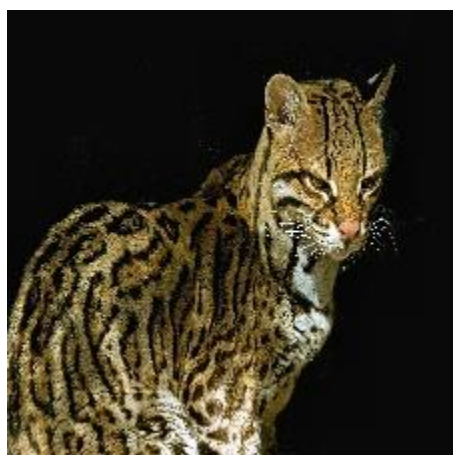
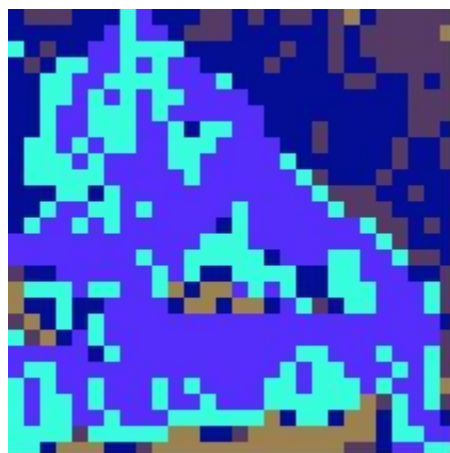
Patch size: 8

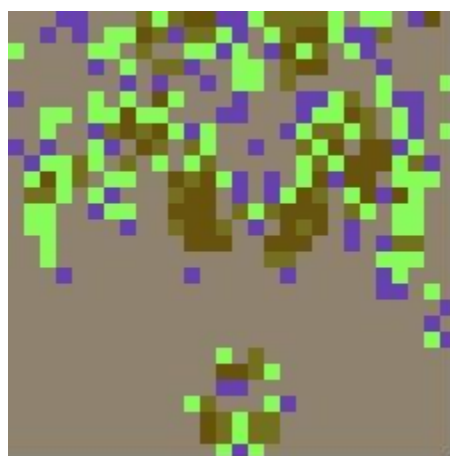
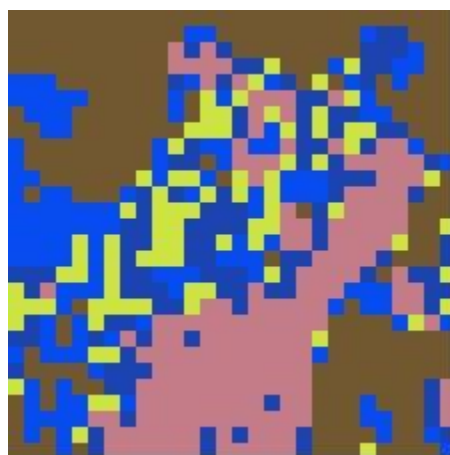
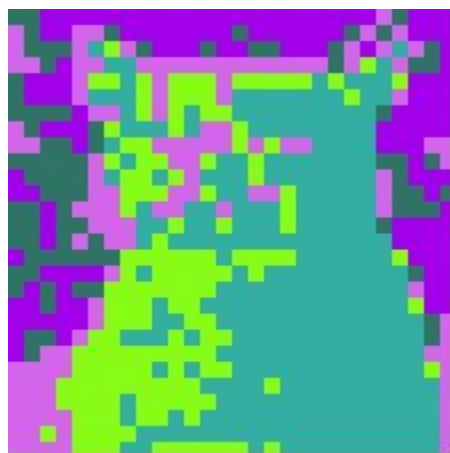
Number of clusters: 5

Heuristic function used: euclidian distance









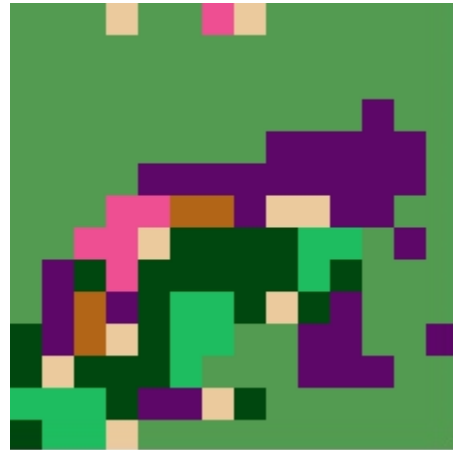
Test #7:

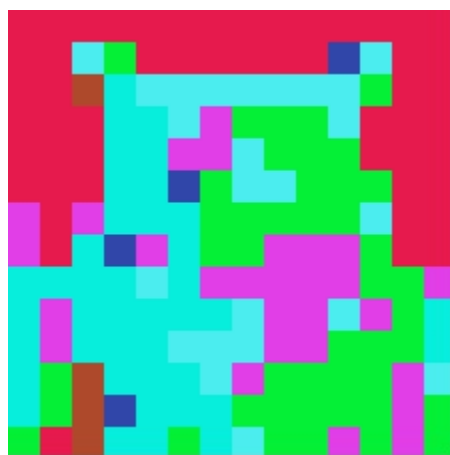
Iterations: 40

Patch size: 16

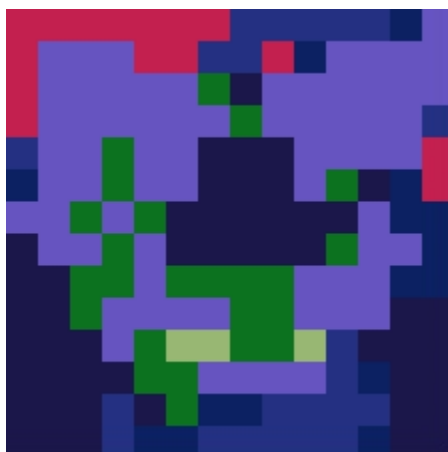
Number of clusters: 7

Heuristic function used: cosine similarity distance









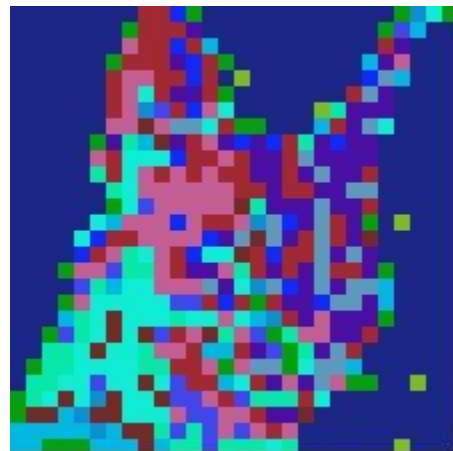
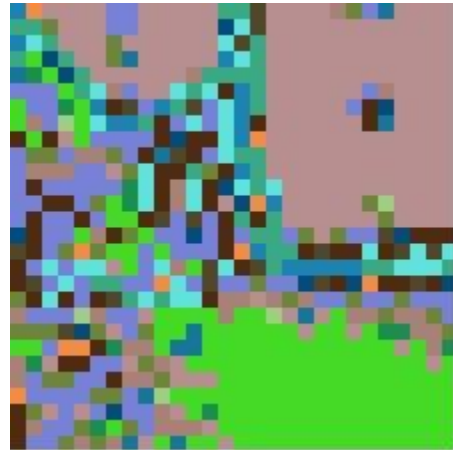
Test #8:

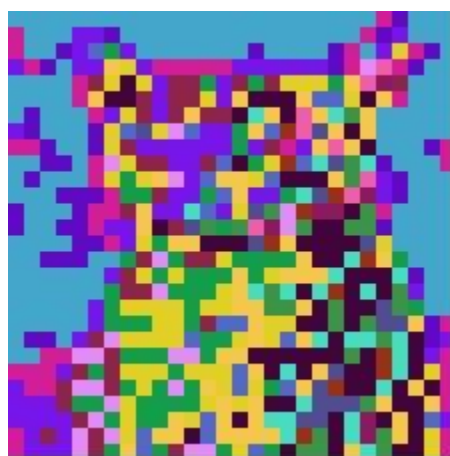
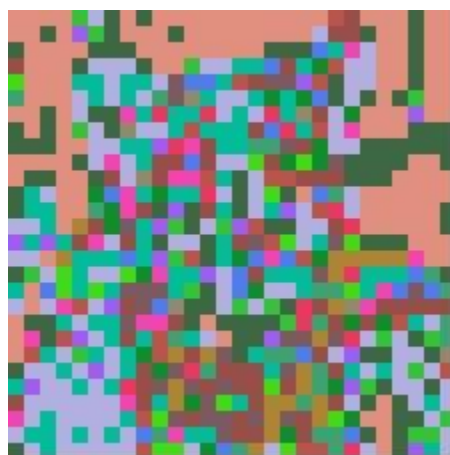
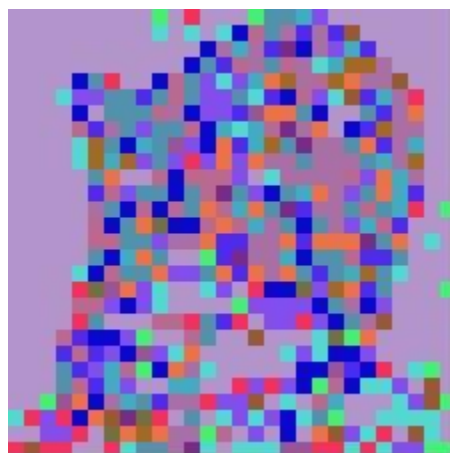
Iterations: 19

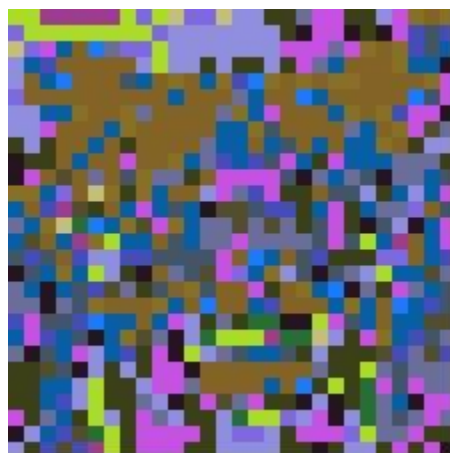
Patch size: 8

Number of clusters: 17

Heuristic function used: euclidian distance









Test #9:

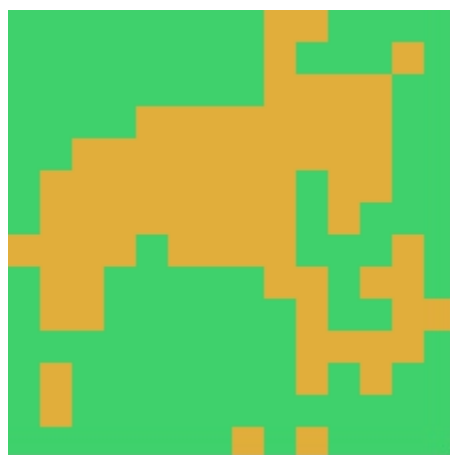
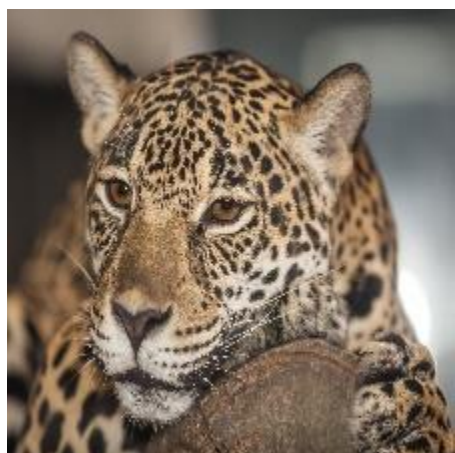
Iterations: 18

Patch size: 16

Number of clusters: 2

Heuristic function used: euclidian distance

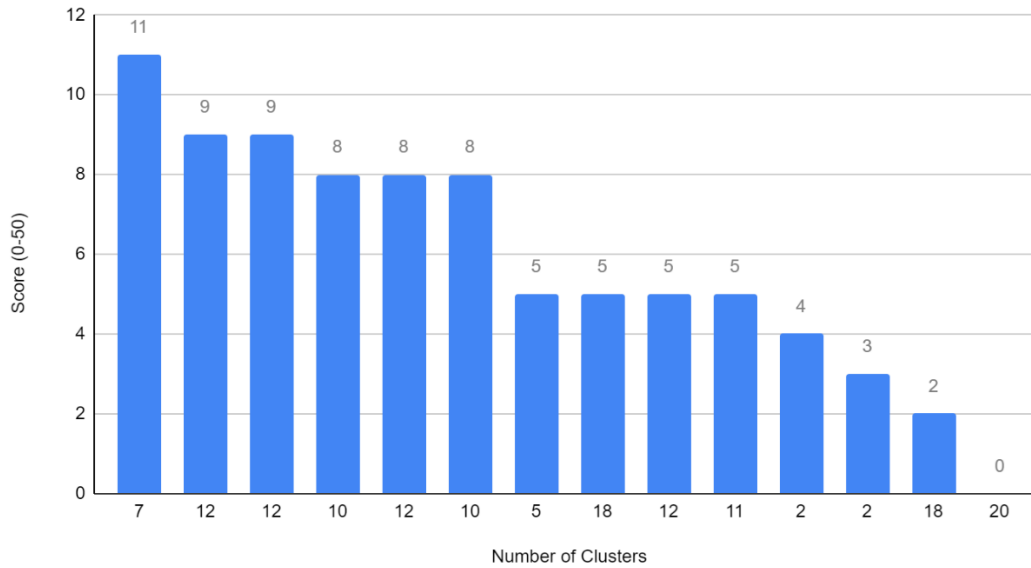




În continuare, vom reprezenta modul în care variația parametrilor influențează rezultatele:

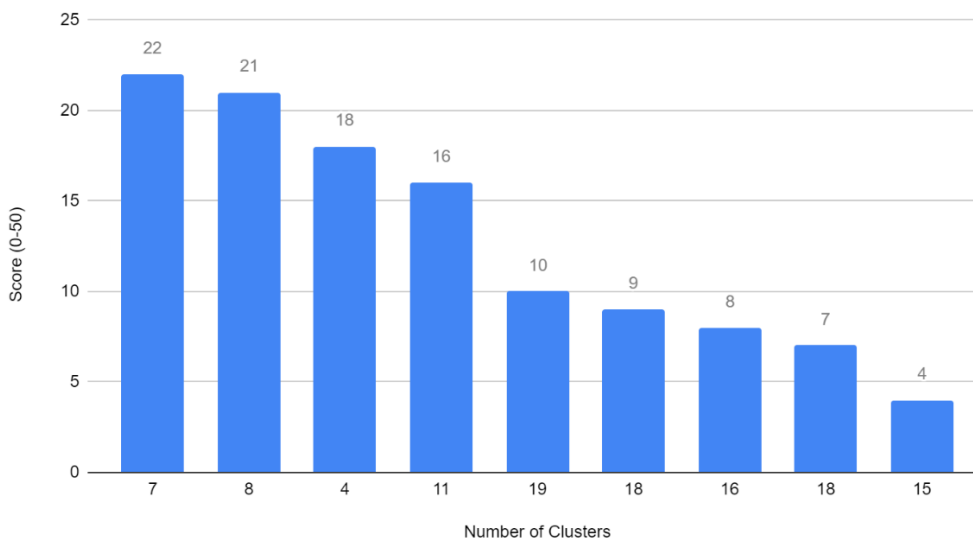
### 1. Euristică Cosine, Patch size 16

Score - Clusters (Heuristic = C, PatchSize = 16)



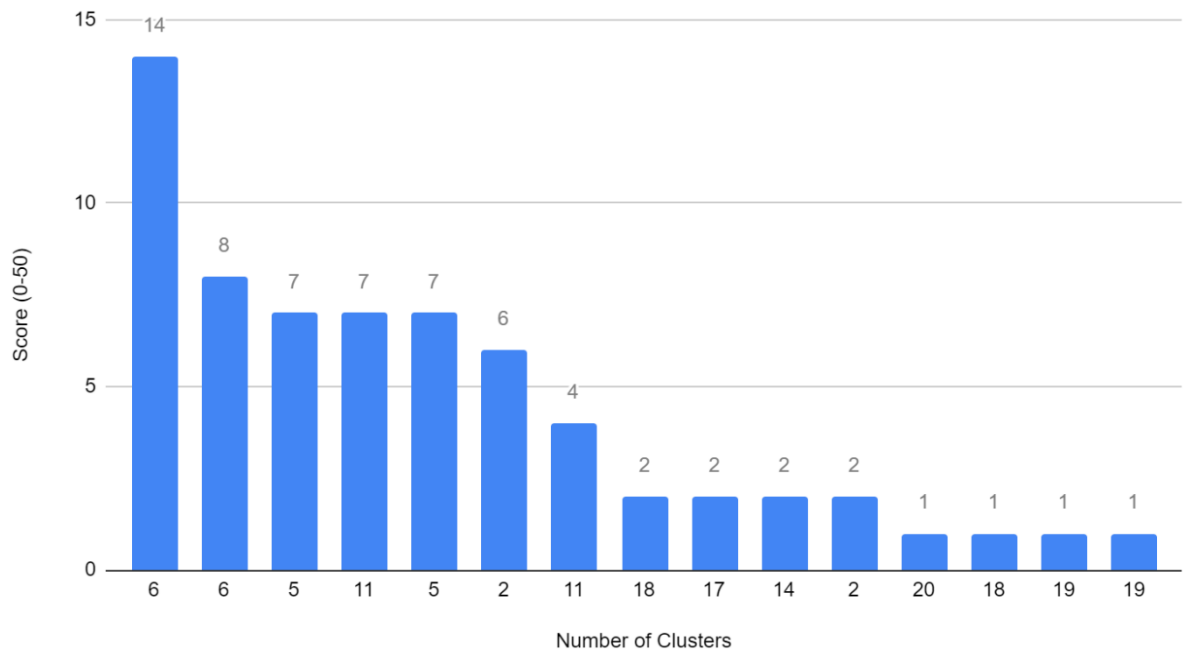
### 2. Euristică Cosine, Patch size 8

Score - Clusters (Heuristic = C, PatchSize = 8)



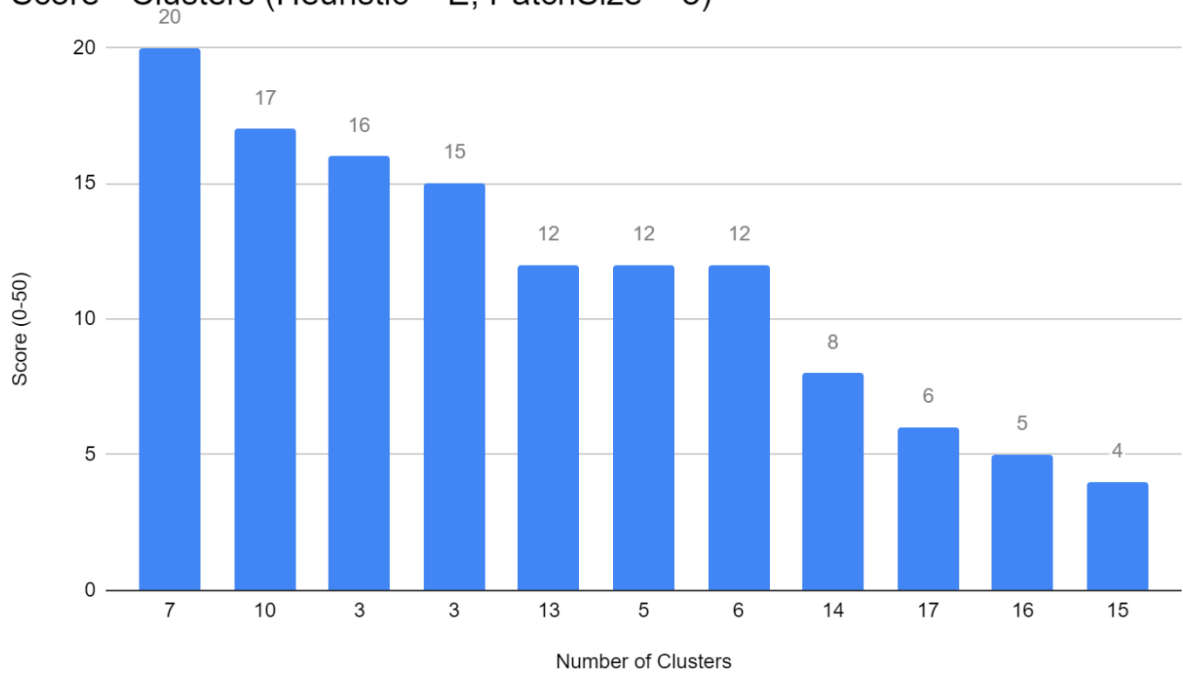
### 3. Euristică Euler, Patch size 16

Score - Clusters (Heuristic = E, PatchSize = 16)



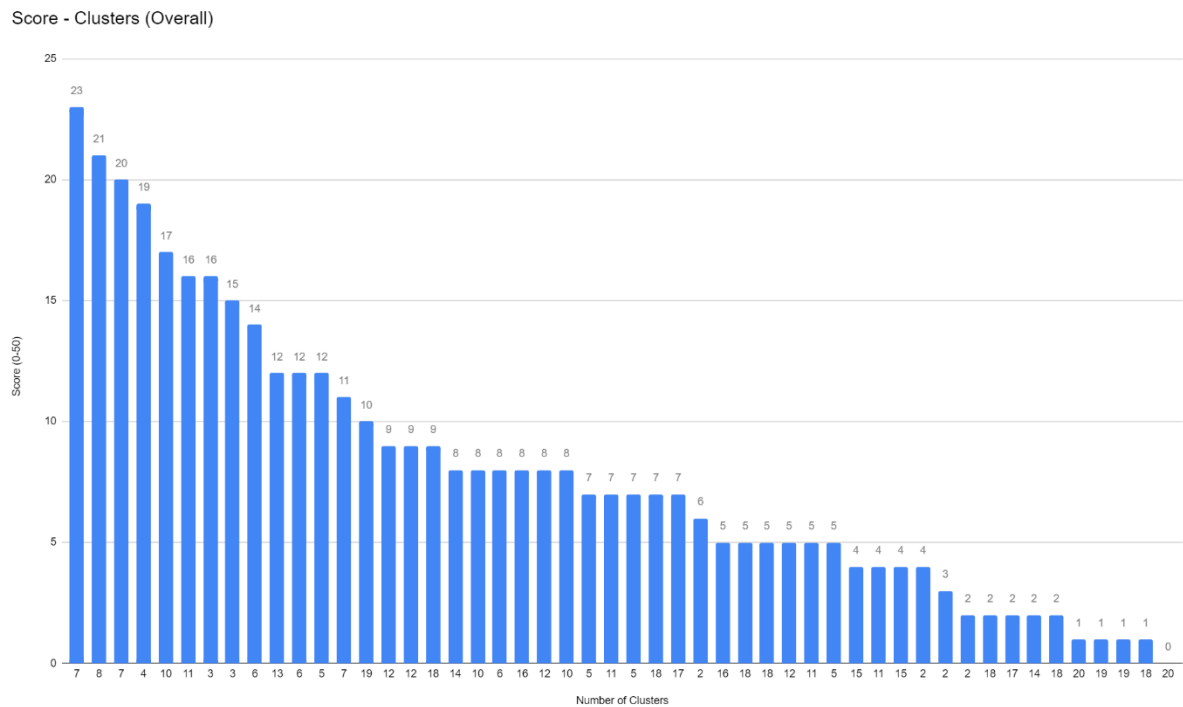
### 4. Euristică Euler, Patch size 8

Score - Clusters (Heuristic = E, PatchSize = 8)





## 5. Overall



În concluzie, observăm că, pentru un rezultat bun, numărul de clustere tinde spre 7, euristică folosită trebuie să fie Cosine Similarity și dimensiunea patchului ar trebui să fie 8.