

# **SPRAWOZDANIE V**

**METODY OBLICZENIOWE W NAUCE I TECHNICE**  
**SINGULAR VALUE DECOMPOSITION**

OCR



**DAWID BIAŁKA**

2019/2020

## Zadanie 1 Analiza obrazów

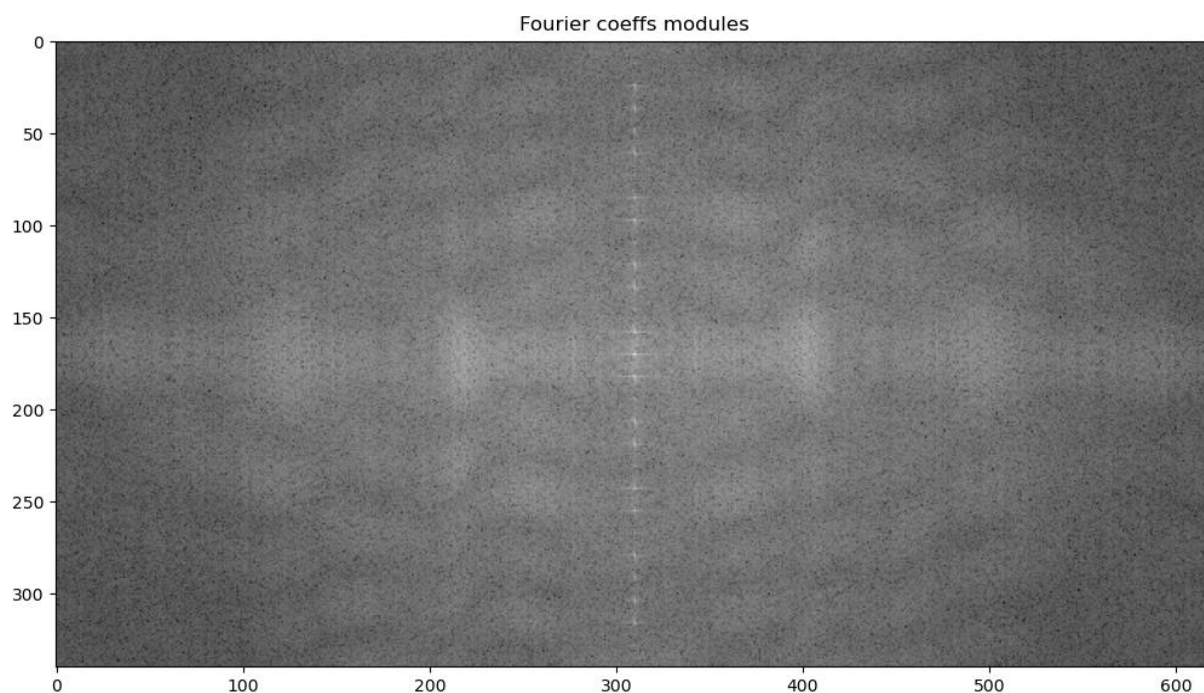
1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u v  
w x y z . , ? ! lorem ipsum dolor sit amet, consectetur  
adipiscing elit. maecenas vel venenatis tellus.  
vivamus ultricies enim libero, at imperdiet nisl feugiat  
eu. nunc sed consequat elit. cras vitae leo eget turpis  
fringilla egestas. nam venenatis, diam at rutrum  
elementum, erat dui vehicula nunc, vitae sollicitudin  
quam eros id ante. nulla scelerisque leo eu neque  
imperdiet molestie. proin consectetur tempus dapibus.  
donec ultricies erat nulla, sit amet luctus arcu  
vulputate vitae. quisque ac congue justo. proin sed  
tincidunt augue, eget pulvinar nisl.

Rys 1. Tekst (jako obraz) wykorzystany do wyszukiwania wzorca. Czcionka - Roboto-Slab-Sherif

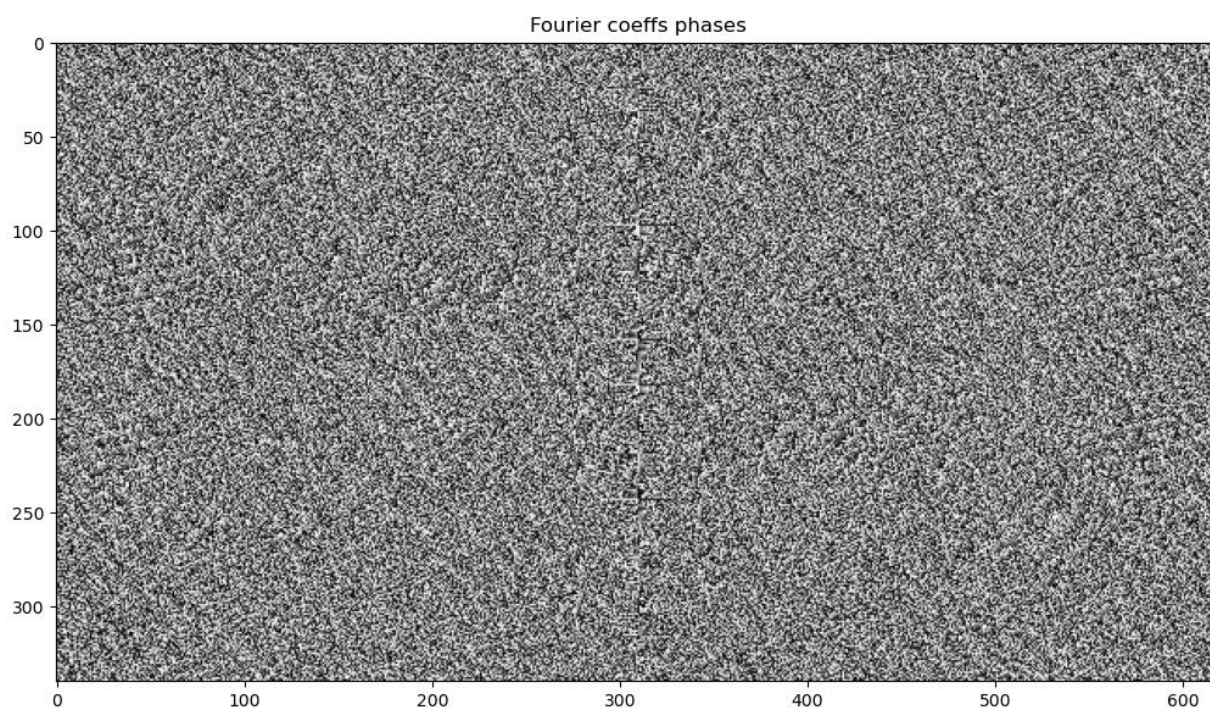
Obracamy kolory na obrazie i transformujemy do domeny częstotliwościowej za pomocą DFT oraz wyliczamy wartości fazy i modułu współczynników Fouriera.

```
freq_array = np.fft.fftshift(np.fft.fft2(array_image))  
  
mods = np.absolute(freq_array)  
mods = np.log(mods)  
phases = np.angle(freq_array)
```

Moduł logarytmujemy, gdyż bez tego otrzymujemy całe czarne tło.



Rys 2. Wartości modułu współczynników Fouriera dla rys. 1.



Rys 3. Wartości fazy współczynników Fouriera dla rys. 1.

Dla rys. 2 możemy zaobserwować charakterystyczne pierścienie rozchodzące się od wewnątrz obrazka i jedną prostą, pionową linię na środku obrazu. W przypadku rys. 3 mamy obraz przypominający ekran telewizora nie podpiętego do sygnału.

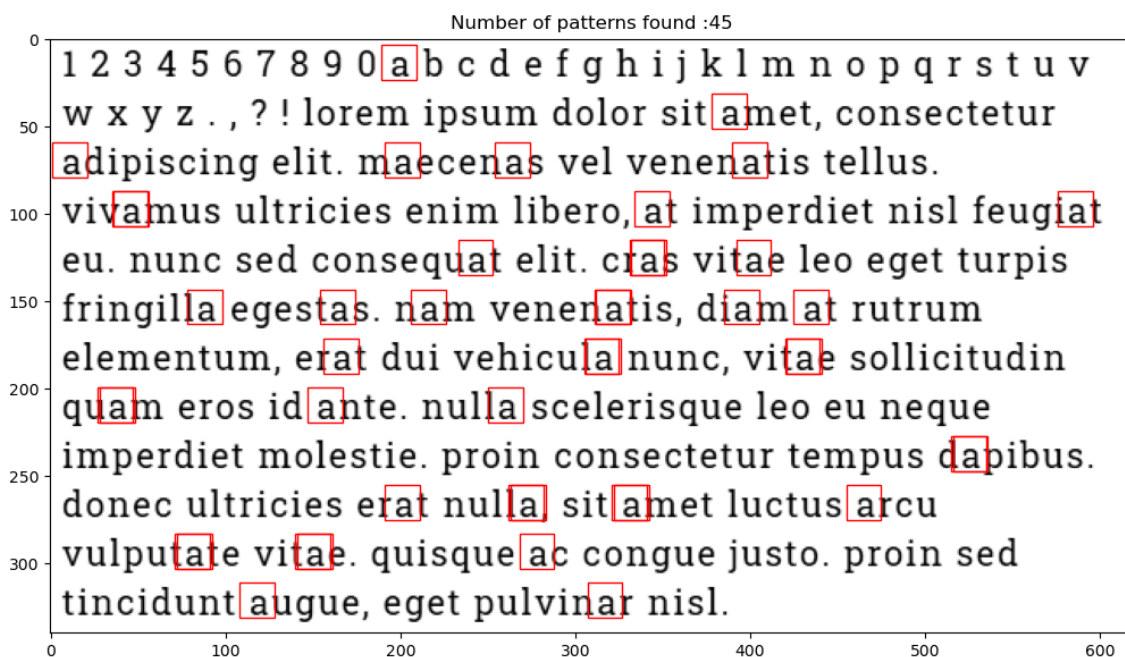
Kolejny krok to wyszukanie wzorca na obrazie. W tym celu wyliczamy korelację pomiędzy obrazem z tekstem a wzorcem a następnie odcinamy wartości, które są mniejsze niż maksymalna wartość korelacji razy pewien współczynnik. Aby znaleźć wzorec, iterujemy po tablicy korelacji i gdy wartość jest różna od zera, w tym punkcie na obrazie występuje wzorec.

```
image_array = np.asarray(loader_image)
pattern_array = np.asarray(loader_pattern)
pattern_array_rotated = np.rot90(pattern_array, 2)

pattern_fft = np.fft.fft2(pattern_array_rotated, np.shape(image_array))
image_fft = np.fft.fft2(image_array)

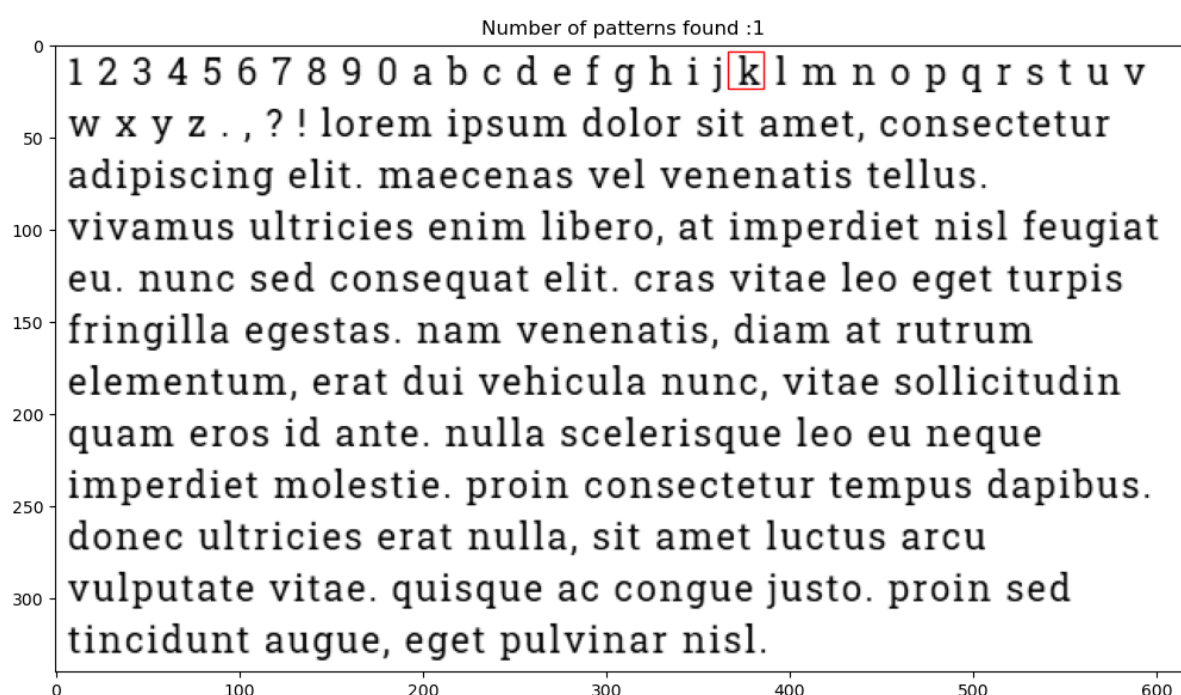
correlation_array = np.real(np.fft.ifft2(np.multiply(pattern_fft, image_fft)))
max = np.max(correlation_array)
for i in range(np.shape(correlation_array)[0]):
    for j in range(np.shape(correlation_array)[1]):
        if(correlation_array[i,j] < max * coeff):
            correlation_array[i,j] = 0
```

Pierwszym analizowanym obrazem będzie kawałek tekstu wygenerowanego na stronie <https://www.lipsum.com/>. Będziemy szukać i zaznaczać kolejno literę „a” i „k”.



Rys. 4 Wyszukiwanie litery „a” na obrazie z tekstem, czcionka – „Roboto-Slab-Sherif”

Liter „a” na tym obrazku jest 34, a widzimy, że program znalazł 45 wystąpień wzorca. Jeśli dobrze się przyjrzymy to w niektórych miejscach litera „a” jest zaznaczona kilka razy. Jest to spowodowane tym, że licząc korelację nie otrzymamy dokładnie jednego punktu, w którym znajduje się wzorec lecz pewien obszar z wysokimi wartościami korelacji. Z tego powodu w następnym zadaniu, gdzie musimy zamienić tekst z obrazu na normalny tekst musimy usuwać te duplikaty.



Rys. 5 Wyszukiwanie litery „k” na obrazie z tekstem, czcionka – „Roboto-Slab-Sherif”

W przypadku litery „k” otrzymaliśmy tylko jedno wystąpienie wzorca, zatem nie zawsze występują duplikaty.

Kolejny obraz to ławica ryb, a wzorcem będzie pewien wycięty fragment ryby.

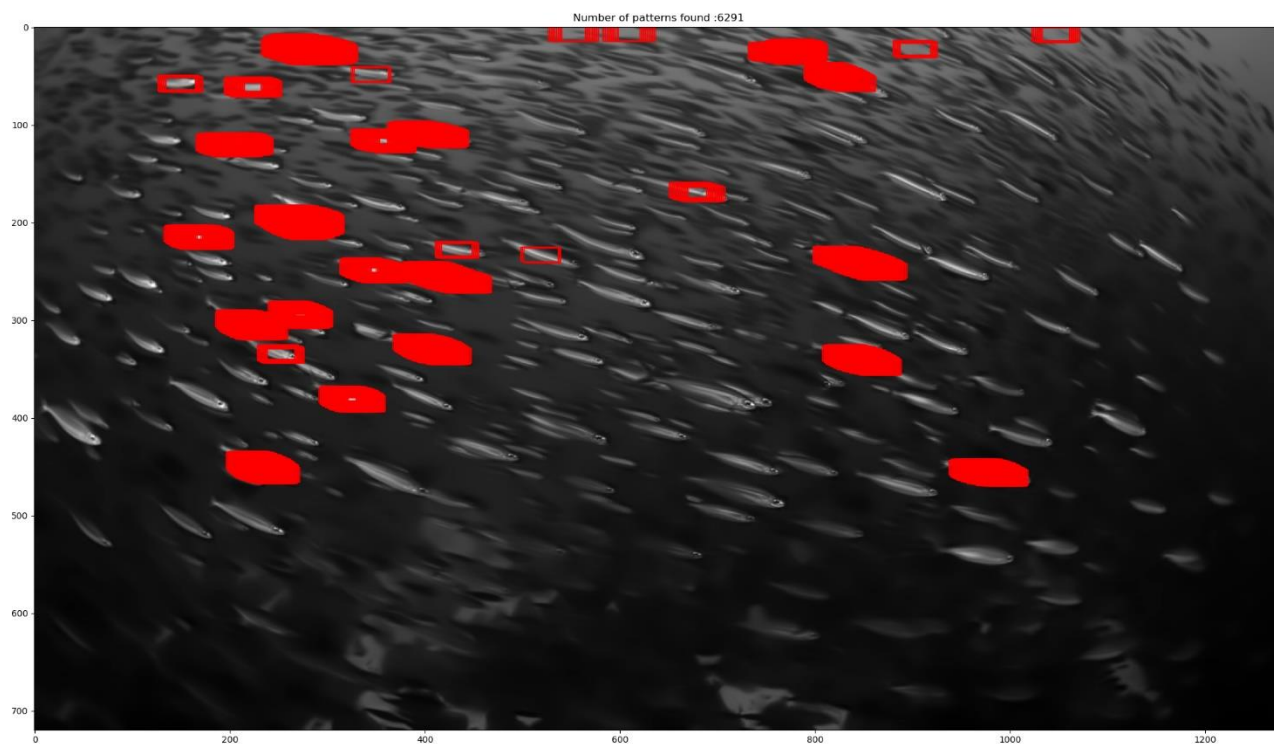




Rys. 6 Obraz z ławicą ryb użyty do znajdowania wzorca



Rys. 7 Użyty wzorzec (sztucznie powiększony w sprawozdaniu, aby go było lepiej widać)



Rys. 8 Obraz z ławicą ryb wraz z nałożonymi ramkami reprezentującymi znaleziony wzorzec.

Widzimy, że w przypadku bardziej skomplikowanych kształtów nie otrzymujemy pików w tablicy korelacji, lecz wartości te są bardziej zbliżone do siebie i przez to otrzymujemy wiele wystąpień wzorca w jednym miejscu.

## Zadanie 2 OCR

Program będzie zamieniał obraz w tekst dla dwóch czcionek : Open-Sans – bezszeryfowa i Roboto-Slab-Sherif – szeryfowa. Tekst składa się wyłącznie z małych liter łacińskich, wszystkie są tego samego rozmiaru i należą do tej samej czcionki, cyfr od 0 do 9 i znaków specjalnych: przecinek, kropka, pytnajnik, wykrzyknik. Tekst może składać się z wielu linii, a kolejne znaki mogą być oddzielone białymi znakami. Możemy wybrać, aby program wypisał liczbę znalezionych znaków, liczbę znaków w oryginalnym tekście oraz procent poprawnie rozpoznanych znaków. Dodatkowo, obraz jest poddawany redukcji szumu wykorzystując funkcję z biblioteki OpenCV.

```
image_array = cv2.fastNlMeansDenoising(np.asarray(image))
```

Program umożliwia również szczytywanie tekstu z obróconego obrazka z tekstem.

Działanie programu opiera się na znalezieniu położenia wszystkich liter na obrazie wykorzystując tą samą metodę co w zadaniu pierwszym. Za to odpowiedzialne są funkcje `compute_correlation()`, `match_all_chars()` i `add_chars()`. Istotne jest to, aby szukać znaków w odpowiedniej kolejności i wszystkie znalezione znaki usuwać z obrazka. Jest to ważne, ponieważ litera b może bardzo łatwo zostać zakwalifikowana jako litera „o”, gdyż kształt litery „o” występuje w kształcie litery „b”. Również ważne jest ustawienie wartości współczynnika `coeff` dla każdej litery, od którego zależy, jak bardzo odcinamy wartości w tablicy korelacji względem maksimum. Wartości te i kolejność wyszukiwani znaków została wyznaczona metodą prób i błędów dla obu czcionek na kilku różnych obrazach. Następnie należy ustawić wszystkie znaki należące do jednej linii, a później dla konkretnych linii ustawić je w odpowiedniej kolejności, takiej, w jakiej występowały w oryginalnym tekście. Realizuje to funkcja `convert_to_text()`.

Wynik działania programu dla kilku przypadków testowych:



1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u v  
w x y z . , ? ! lorem ipsum dolor sit amet, consectetur  
adipiscing elit. maecenas vel venenatis tellus.  
vivamus ultricies enim libero, at imperdiet nisl feugiat  
eu. nunc sed consequat elit. cras vitae leo eget turpis  
fringilla egestas. nam venenatis, diam at rutrum  
elementum, erat dui vehicula nunc, vitae sollicitudin  
quam eros id ante. nulla scelerisque leo eu neque  
imperdiet molestie. proin consectetur tempus dapibus.  
donec ultricies erat nulla, sit amet luctus arcu  
vulputate vitae. quisque ac congue justo. proin sed  
tincidunt augue, eget pulvinar nisl.

Rys. 9 Obraz do transformacji do tekstu – przykład pierwszy – czcionka „Roboto-Slab-Sherif”

Otrzymany tekst:

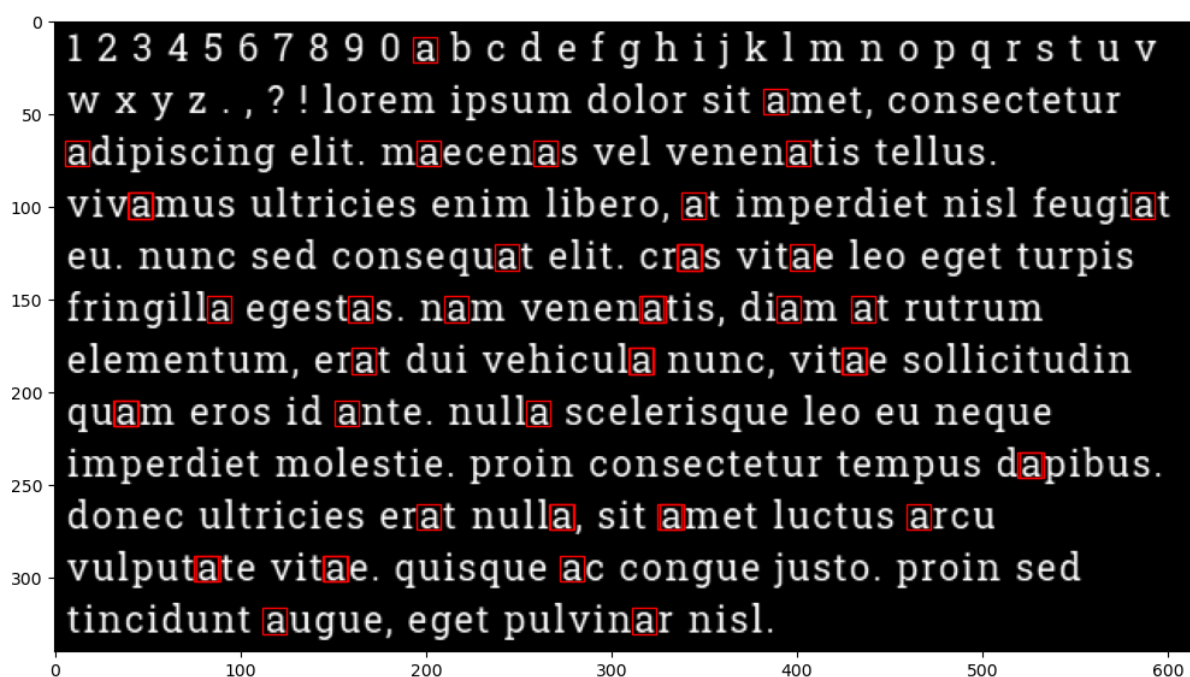
1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u v  
w x y z . , ? ! lorem ipsum dolor sit amet, consectetur  
adipiscing elit. maecenas vel venenatis tellus.  
vivamus ultricies enim libero, at imperdiet nisl feugiat  
eu. nunc sed consequat elit. cras vitae leo eget turpis  
fringilla egestas. nam venenatis, diam at rutrum  
elementum, erat dui vehicula nunc, vitae sollicitudin  
quam eros id ante. nulla scelerisque leo eu neque  
imperdiet molestie. proin consectetur tempus dapibus.  
donec ultricies erat nulla, sit amet luctus arcu  
vulputate vitae. quisque ac congue justo. proin sed  
tincidunt augue, eget pulvinar nisl.

# Raport poprawnie zeskanowych znaków:

Char	Found	Original	Ratio
a	34	34	100.0 %
d	14	14	100.0 %
e	64	64	100.0 %
f	3	3	100.0 %
g	9	9	100.0 %
q	7	7	100.0 %
b	3	3	100.0 %
h	2	2	100.0 %
4	1	1	100.0 %
t	43	43	100.0 %
p	12	12	100.0 %
j	2	2	100.0 %
s	33	33	100.0 %
k	1	1	100.0 %
m	18	18	100.0 %
n	32	32	100.0 %
u	41	41	100.0 %
r	23	23	100.0 %
6	1	1	100.0 %
o	18	18	100.0 %
l	30	30	100.0 %
w	1	1	100.0 %
c	22	22	100.0 %
x	1	1	100.0 %
y	1	1	100.0 %
v	12	12	100.0 %
z	1	1	100.0 %
0	1	1	100.0 %
8	1	1	100.0 %

9	1	1	100.0 %
1	1	1	100.0 %
i	47	47	100.0 %
2	1	1	100.0 %
3	1	1	100.0 %
5	1	1	100.0 %
7	1	1	100.0 %
exclamation	1	1	100.0 %
question	1	1	100.0 %
comma	8	8	100.0 %
dot	12	12	100.0 %

Działanie programu krok po kroku, wraz z usuwaniem znaków z obrazu. (pokazane tylko pierwsze trzy obrazki, gdyż jest ich 40 dla każdego obrazu, znajdują się one w folderach step\_by\_step)



0 1 2 3 4 5 6 7 8 9 0 b c d e f g h i j k l m n o p q r s t u v  
 50 w x y z . , ? ! lorem ipsum d olor sit met, consecetur  
 100 d ipiscing elit. m ecen s vel venen tis tellus.  
 150 viv mus ultricies enim libero, t imper d iet nisl feugi t  
 200 eu. nunc se d consequ t elit. cr s vit e leo eget turpis  
 250 fringill egest s. n m venen tis, d i m t rutrum  
 300 elementum, er t d ui vehicul nunc, vit e sollicitu d in  
 qu m eros id nte. null scelerisque leo eu neque  
 imper d iet molestie. proin consecetur tempus d pibus.  
 donec ultricies er t null , sit met luctus rcu  
 vulput te vit e. quisque c congue justo. proin se d  
 tinci d unt ugue, eget pulvin r nisl.

0 1 2 3 4 5 6 7 8 9 0 b c e f g h i j k l m n o p q r s t u v  
 50 w x y z . , ? ! lore m ipsum olor sit met, consecetur  
 100 ipiscing elit. m ecen s vel venen tis tellus.  
 150 viv mus ultricies enim libero, t imper iet nisl feugi t  
 200 eu. nunc se consequ t elit. cr s vit e leo eget turpis  
 250 fringill egest s. n m venen tis, i m t rutrum  
 300 elementum, er t ui vehicul nunc, vit e sollicitu in  
 qu m eros i nte. null scelerisque leo eu neque  
 imper iet molestie. proin consecetur tempus pibus.  
 onec ultricies er t null , sit met luctus rcu  
 vulput te vit e. quisque c congue justo. proin se  
 tinci unt ugue, eget pulvin r nisl.

Rys. 10 Kolejne etapy działania programu dla przykładu pierwszego

1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u v  
w x y z . , ? ! in et malesuada ipsum! aliquam commodo  
augue, ut sapien tincidunt dictum? proin 6quis lectus  
sit amet diam luctus vehicula. praesent eleifend  
faucibus nibh, mollis suscipit nulla tincidunt vel.  
curabitur 349pharetra odio at odio auctor, id pulvinar  
felis semper. phasellus eros lorem, bibendum ac  
elementum id, scelerisque vitae 765orci! fusce pulvinar  
magna et erat faucibus sagittis. aliquam aliquam  
vestibulum libero, vitae gravida quam condimentum  
et. 123nunc hendrerit imperdiet urna, non egestas  
lectus ullamcorper non. mauris 990 non scelerisque  
mauris? maecenas aliquam sodales commodo. donec  
vitae arcu mi.

Rys. 10 Obraz do transformacji do tekstu – przykład drugi – czcionka „Roboto-Slab-Sherif”

Otrzymany tekst:

1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u v  
w x y z . , ? ! in et malesuada ipsum! aliquam commodo  
augue, ut sapien tincidunt dictum? proin 6quis lectus  
sit amet diam luctus vehicula. praesent eleifend  
faucibus nibh, mollis suscipit nulla tincidunt vel.  
curabitur 349pharetra odio at odio auctor, id pulvinar  
felis semper. phasellus eros lorem, bibendum ac  
elementum id, scelerisque vitae 765orci! fusce pulvinar  
magna et erat faucibus sagittis. aliquam aliquam  
vestibulum libero, vitae gravida quam condimentum  
et. 123nunc hendrerit imperdiet urna, non egestas  
lectus ullamcorper non. mauris 990 non scelerisque  
mauris? maecenas aliquam sodales commodo. donec  
vitae arcu mi.

# Raport poprawnie zeskanowych znaków:

Char	Found	Original	Ratio
a	49	49	100.0 %
d	20	20	100.0 %
e	53	53	100.0 %
f	6	6	100.0 %
g	6	6	100.0 %
q	9	9	100.0 %
h	6	6	100.0 %
b	9	9	100.0 %
4	2	2	100.0 %
t	33	33	100.0 %
p	13	13	100.0 %
j	1	1	100.0 %
s	35	35	100.0 %
k	1	1	100.0 %
m	31	31	100.0 %
n	31	31	100.0 %
u	46	46	100.0 %
r	28	28	100.0 %
6	3	3	100.0 %
o	25	25	100.0 %
l	30	30	100.0 %
w	1	1	100.0 %
c	26	26	100.0 %
x	1	1	100.0 %
y	1	1	100.0 %
v	10	10	100.0 %
z	1	1	100.0 %
0	2	2	100.0 %
8	1	1	100.0 %
9	4	4	100.0 %



1	2	2	100.0 %
i	52	52	100.0 %
2	2	2	100.0 %
3	3	3	100.0 %
5	2	2	100.0 %
7	2	2	100.0 %
exclamation	3	3	100.0 %
question	3	3	100.0 %
comma	8	8	100.0 %
dot	9	9	100.0 %

1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u  
 v w x y z . , ? ! quisque lorem quam, ullamcorper et  
 viverra sit amet, semper ut erat. vestibulum ante  
 ipsum primis in faucibus orci luctus et ultrices  
 posuere cubilia curae. nunc varius blandit tincidunt!  
 aliquam porta felis id finibus rutrum. ut et eros enim.  
 cras convallis nulla in bibendum mollis? dui at  
 eleifend sapien, et tincidunt sapien. vestibulum at  
 hendrerit enim. etiam faucibus tempor arcu, ut  
 ultrices mauris tincidunt ut. in interdum  
 condimentum sapien, quis suscipit dui gravida  
 interdum. quisque varius malesuada metus eget  
 molestie. phasellus fringilla, ex eget dignissim  
 vehicula, massa nisl malesuada orci, sit amet aliquam  
 lorem ipsum aliquam elit? in venenatis nulla eu  
 elementum ultricies! proin sed placerat urna.

Rys. 11 Obraz do transformacji do tekstu – przykład trzeci – czcionka „Open-Sans”

Otrzymany tekst:

1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u  
v w x y z . , ? ! quisque lorem quam, ullamcorper et  
viverra sit amet, semper ut erat. vestibulum ante  
ipsum primis in faucibus orci luctus et ultrices  
posuere cubilia curae. nunc varius blandit tincidunt!  
aliquam posuere feiis id finibus rutrum. ut et eros enim.  
cras convallis nulla in bibendum mollis? duis at  
eleifend sapien, et tincidunt sapien. vestibulum at  
hendrerit enim. etiam faucibus tempor arcu, ut  
ultrices mauris tincidunt ut. in interdum  
condimentum sapien, quis suscipit dui gravida  
interdum. quisque varius malesuada metus eget  
molestie. phasellus fringilla, ex eget dignissim  
vehicula, massa nisl malesuada orci, sit amet aliquam  
lorem ipsum aliquam elit? in venenatis nulla eu  
elementum ultricies! proin sed placerat urna.

Raport poprawnie zeskanowanych znaków:

Char	Found	Original	Ratio
a	51	51	100.0 %
d	19	19	100.0 %
e	64	64	100.0 %
f	7	7	100.0 %
g	6	6	100.0 %
q	10	10	100.0 %
b	10	10	100.0 %
h	4	4	100.0 %
t	47	47	100.0 %
4	1	1	100.0 %
0	1	1	100.0 %
j	1	1	100.0 %

s	47	47	100.0 %
k	1	1	100.0 %
m	36	36	100.0 %
p	16	16	100.0 %
n	39	39	100.0 %
l	30	38	78.94736842105263 %
u	61	61	100.0 %
r	35	36	97.22222222222221 %
6	1	1	100.0 %
o	15	15	100.0 %
w	1	1	100.0 %
c	23	23	100.0 %
x	2	2	100.0 %
y	1	1	100.0 %
v	11	11	100.0 %
z	1	1	100.0 %
8	1	1	100.0 %
9	1	1	100.0 %
1	1	1	100.0 %
i	82	74	110.8108108108108 %
2	1	1	100.0 %
3	1	1	100.0 %
5	1	1	100.0 %
7	1	1	100.0 %
exclamation	3	3	100.0 %
question	3	3	100.0 %
comma	10	9	111.11111111111111 %
dot	12	11	109.09090909090908 %

1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u  
v w x y z . , ? ! lorem ipsum dolor sit amet,  
consectetur adipiscing elit. maecenas vel venenatis  
tellus. vivamus ultricies enim libero, at imperdiet nisl  
feugiat eu. nunc sed consequat elit. cras vitae leo  
eget turpis fringilla egestas? nam venenatis, diam at  
rutrum elementum, erat dui vehicula nunc, vitae  
sollicitudin quam eros id ante. nulla scelerisque? leo  
eu neque imperdiet molestie! proin consectetur  
tempus dapibus. donec ultricies erat nulla, sit amet  
luctus arcu vulputate vitae. quisque ac congue justo!  
proin sed tincidunt augue, eget pulvinar nisl.

Rys. 12 Obraz, obrócony do transformacji do tekstu – przykład czwarty – czcionka „Open-Sans”

Otrzymany tekst:

1 2 3 4 5 6 7 8 9 0 a b c d e f g h i j k l m n o p q r s t u  
v w x y z . , ? ! lorem ipsum dolor sit amet,  
consectetur adipiscing elit. maecenas vel venenatis  
tellus. vivamus ultricies enim libero, at imperdiet nisl  
feugiat eu. nunc sed consequat elit. cras vitae leo  
eget turpis fringilla egestas? nam venenatis, diam at  
rutrum elementum, erat dui vehicula nunc, vitae  
solicitudin quam eros id ante. nulla scelerisque? leo  
eu neque imperdiet mollis, tunc! proin consectetur  
tempus dapibus. donec ultricies eu, at nulla, sit amet  
luctus arcu vulputate vitae. quisque ac congue justo!  
proin sed tincidunt augue, eget pulvinar nisl.

Raport poprawnie zeskanowanych znaków:

Char	Found	Original	Ratio
a	34	34	100.0 %
d	14	14	100.0 %
e	61	64	95.3125 %
f	3	3	100.0 %
g	9	9	100.0 %
q	7	7	100.0 %
b	3	3	100.0 %
h	2	2	100.0 %
t	43	43	100.0 %
4	1	1	100.0 %
0	1	1	100.0 %
j	2	2	100.0 %
s	32	33	96.969696969697 %
k	1	1	100.0 %
m	18	18	100.0 %
p	12	12	100.0 %

n	31	32	96.875 %
l	25	30	83.33333333333334 %
u	41	41	100.0 %
r	23	23	100.0 %
6	1	1	100.0 %
o	18	18	100.0 %
w	1	1	100.0 %
c	24	22	109.09090909090908 %
x	1	1	100.0 %
y	1	1	100.0 %
v	12	12	100.0 %
z	1	1	100.0 %
8	1	1	100.0 %
9	1	1	100.0 %
1	6	1	600.0 %
i	47	47	100.0 %
2	1	1	100.0 %
3	1	1	100.0 %
5	1	1	100.0 %
7	1	1	100.0 %
exclamation	3	3	100.0 %
question	3	3	100.0 %
comma	12	8	150.0 %
dot	12	9	133.33333333333331 %





Rys. 13 Wyprostowany obraz z rys.12, etap wyszukiwania litery „u” – część liter już usunięta