**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU ELEKTROTEHNIČKI FAKULTET**

**Sveučilišni studij**

**METODA PODUDARANJA SUSJEDSTVA PIKSELA S MODELOM**

**Vježba 2**

**Antonio Lončar**

**Osijek, 2015.**

# Zadatak i rješenje

Napisati funkciju tako da korisnik pomoću miša može označiti jedno slovo u tekstu na slici nakon čega program automatski označava (uokviruje kvadratićem) sva slična (ista) slova poput označenog slova primjenom metode podudaranja susjedstva piksela s modelom. Dobiveni rezultat treba prikazati u novootvorenom prozoru. Ova funkcija se pokreće klikom na gumb “**Template** **matching**”, a pomoću **Esc** se izlazi iz tog načina rada.

## Rješenje

Korištenjem funkcije matchImage(Mat inImage, Mat roi) dobijemo rezultat sliku na kojoj je prikazana podudaranje označenoga ROI-a sa cijelom slikom.

void matchImage(Mat inputImage, Mat ROI, double thresHold)

{

Mat resultCCORR, resultSQDIFF, resultCCOEFF

Mat imgCCORR, imgSQDIFF, imgCCOFF;

int nCCORR, nSQDIFF, nCCOEFF;

inputImage.copyTo(imgCCORR);

inputImage.copyTo(imgSQDIFF);

inputImage.copyTo(imgCCOFF);

cvtColor(inputImage, inputImage, CV\_BGR2GRAY);

cvtColor(ROI, ROI, CV\_BGR2GRAY);

matchTemplate(inputImage, ROI, resultCCORR, CV\_TM\_CCORR\_NORMED);

matchTemplate(inputImage, ROI, resultSQDIFF, CV\_TM\_SQDIFF\_NORMED);

matchTemplate(inputImage, ROI, resultCCOEFF, CV\_TM\_CCOEFF\_NORMED);

imshow("WindowRes resultCCOEFF", resultCCOEFF); // najbolji

imshow("WindowRes resultSQDIFF", resultSQDIFF); // inverzan

imshow("WindowRes resultCCORR", resultCCORR);

/\* Prolazi kroz obrađenu sliku i pronalazi tock gdje je vrijednost veca o 0.85\*/

int n = 0;

for (int i = 0; i < resultCCORR.cols; i++) //stupci

{

for (int j = 0; j < resultCCORR.rows; j++)

{

if (resultCCORR.at<float>(j, i) > thresHold + (1 - thresHold)/2)

{

rectangle(imgCCORR, Rect(i, j, ROI.cols, ROI.rows), Scalar(0, 0, 255));

nCCORR++;

}

if (resultSQDIFF.at<float>(j, i) < 0.95-thresHold)

{

rectangle(imgSQDIFF, Rect(i, j, ROI.cols, ROI.rows), Scalar(0, 0, 255));

nSQDIFF++;

}

if (resultCCOEFF.at<float>(j, i) > thresHold)

{

rectangle(imgCCOFF, Rect(i, j, ROI.cols, ROI.rows), Scalar(0, 0, 255));

nCCOEFF++;

}

}

}

imshow("WindowRes imgCCORR", imgCCORR);

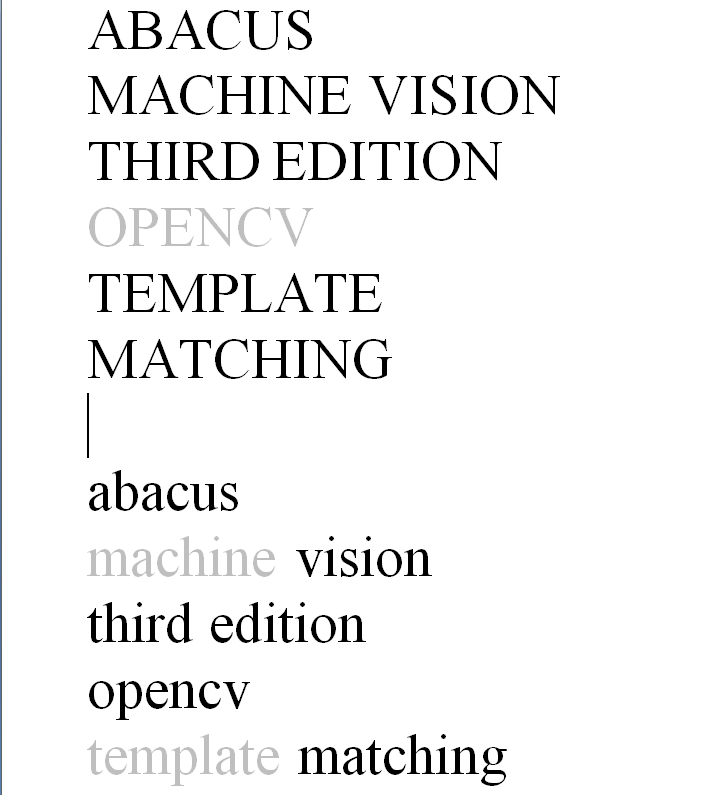
imshow("WindowRes imgSQDIFF", imgSQDIFF);

imshow("WindowRes imgCCOFF", imgCCOFF);

}

Funkcija ima tri parametra. Sliku na kojoj se nalazi roi, roi i prag podudaranja. Varijbale nCCORR, nSQDIFF i nCCOFF zadrze broj pronađenih slova na slici. Glavna funkcija koja je zaslužna za određivanje podudranja je matchTemplate () funkcija. Metode traženja sličnosti su sljedeći: CV\_TM\_SQDIFF , CV\_TM\_CCORR i CV\_TM\_CCOEFF , te njihove normalizirane verzije: CV\_TM\_SQDIFF\_NORMED, CV\_TM\_CCORR\_NORMED i CV\_TM\_CCOEFF\_NORMED

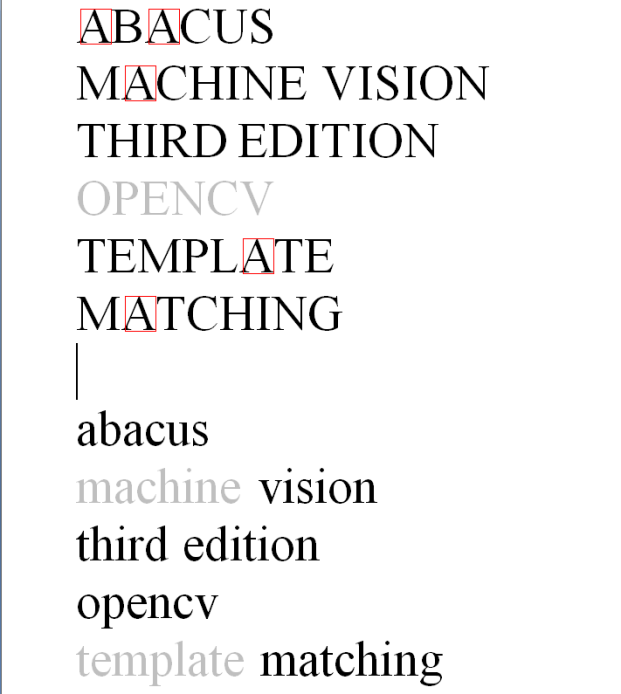
Na sljedećim slikama je dan primjer traženja slova 'A' na slici 1.1., korištene su sve tri normalizirane metode traženja. Vrijednost tresholda za metodu CCOEFF je 0.88, za CCORR je 0.94, a za SQDIFF je 0.12.



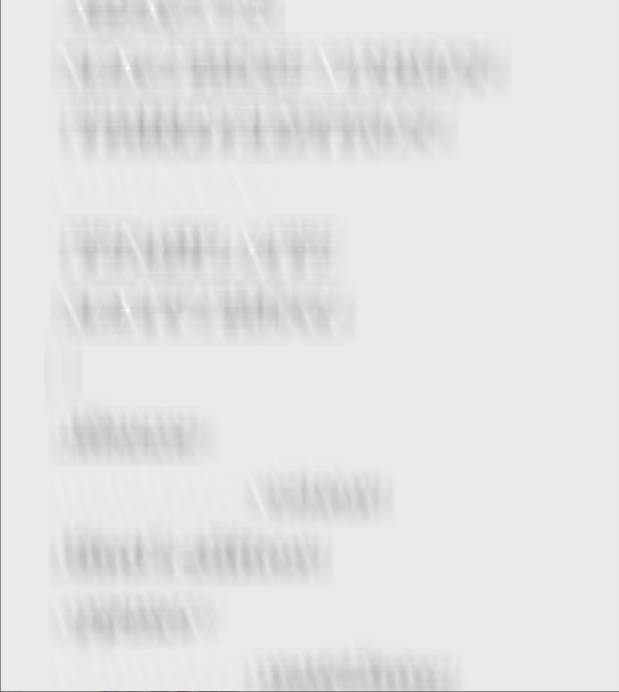
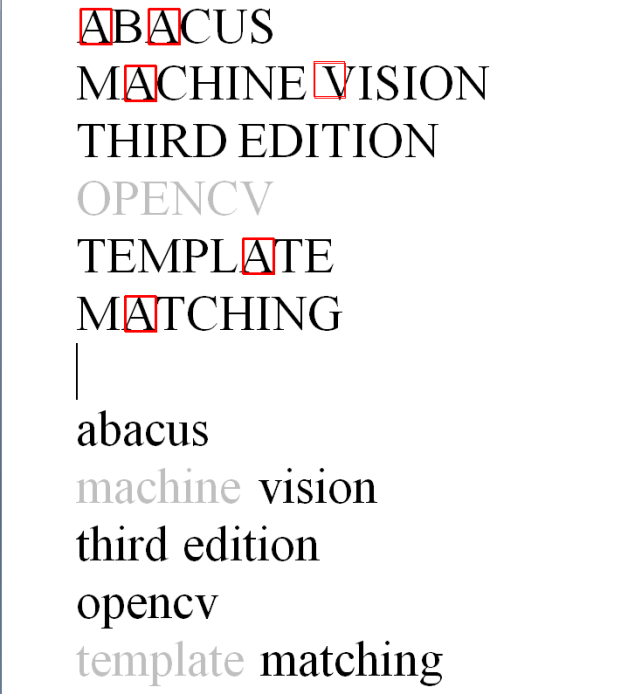
Sl. 1.1. Zadana slika.

C:\Users\XXX\Desktop\slike lv2\roi.bmp

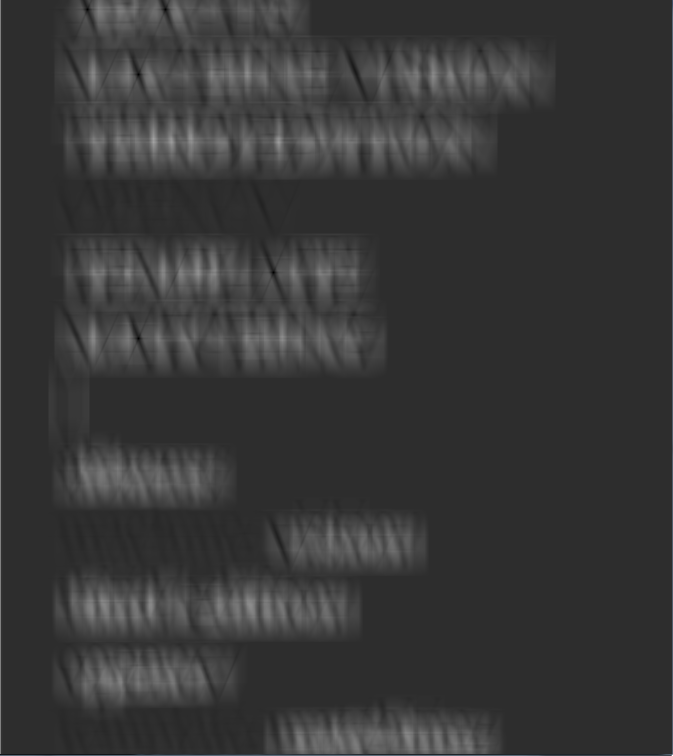
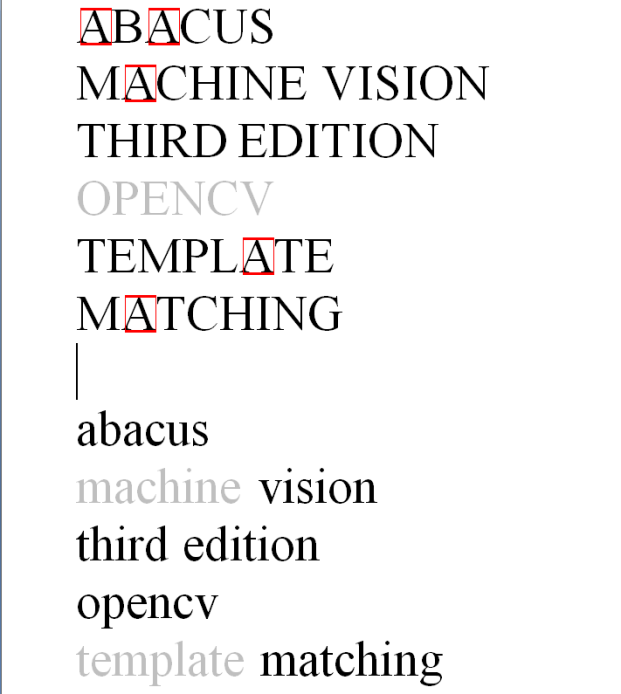
Sl. 1.2. Slovo koje se trazi.

Sl. 1.3. Rezultat dobiven CCOEFF metodom

Sl. 1.4. Rezultat dobiven CCORR metodom

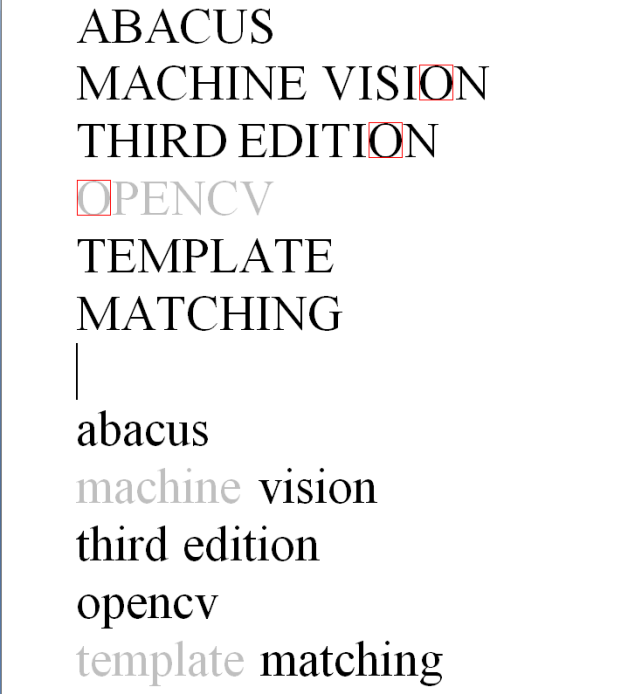
Sl. 1.5. Rezultat dobiven SQDIFF metodom

Iz dobivenih slika možemo vidjeti da kod CCOEFF i CCORR metoda mjesto najvjerojatnijeg pojavljivanja ROI-a je prikazano globalnim maksimumom, dok kod SQDIFF metode globalnim minimumom.

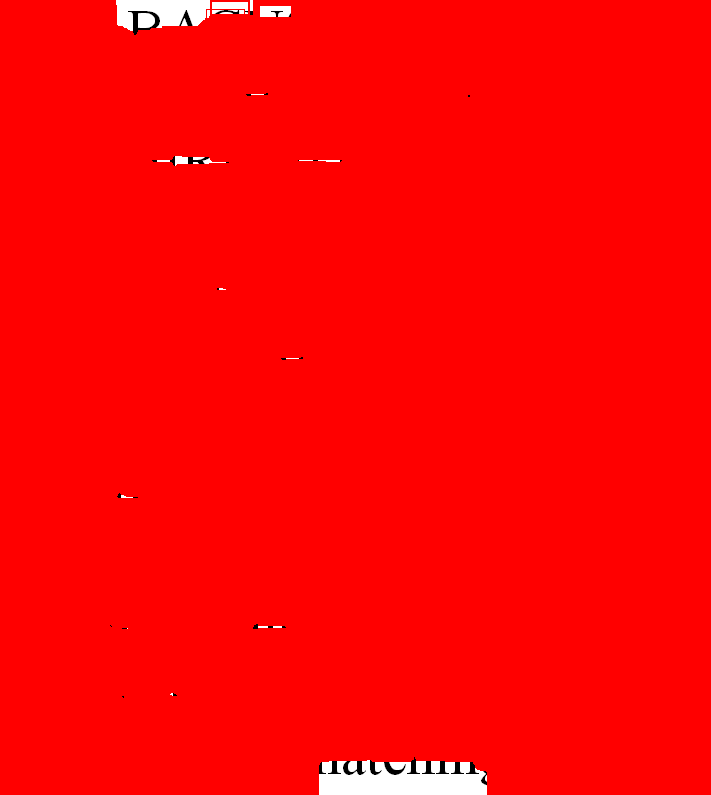
Sljedeći primjer je slovo 'O' koje nije potpuno crno već je sive boje. Treshold vrijednost je ista kao i u prošlome slućaju.

C:\Users\XXX\Desktop\slike lv2\roi.bmp

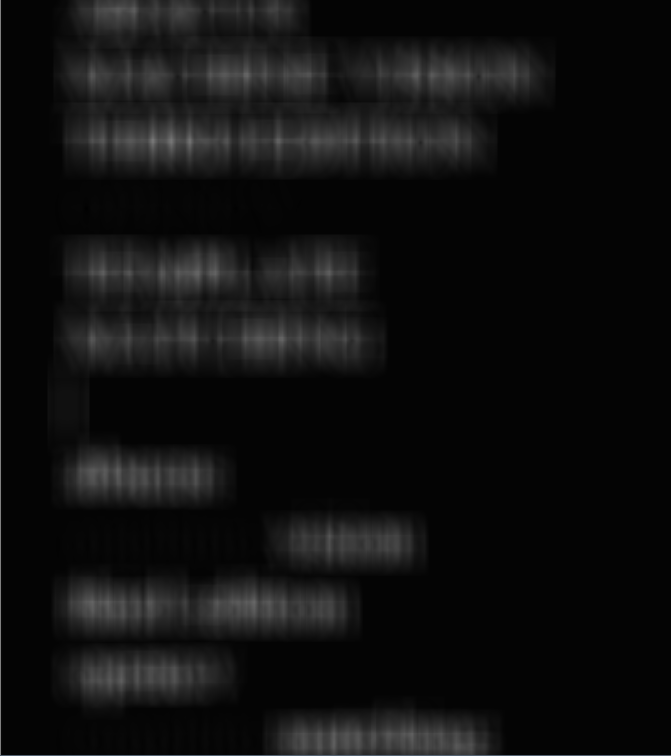
Sl. 1.6. Slovo koje se traži.

Sl. 1.4. Rezultat dobiven CCOEFF metodom

Sl. 1.5. Rezultat dobiven CCORR metodom

Sl. 1.6. Rezultat dobiven SQDIFF metodom

Iz dobivenih slika možemo vidjeti da metoda CCOEFF najbolje obavlja pronalazi zadano slovo.