

# Vzporedni in porazdeljeni sistemi in algoritmi: Vaje

OpenCL

---

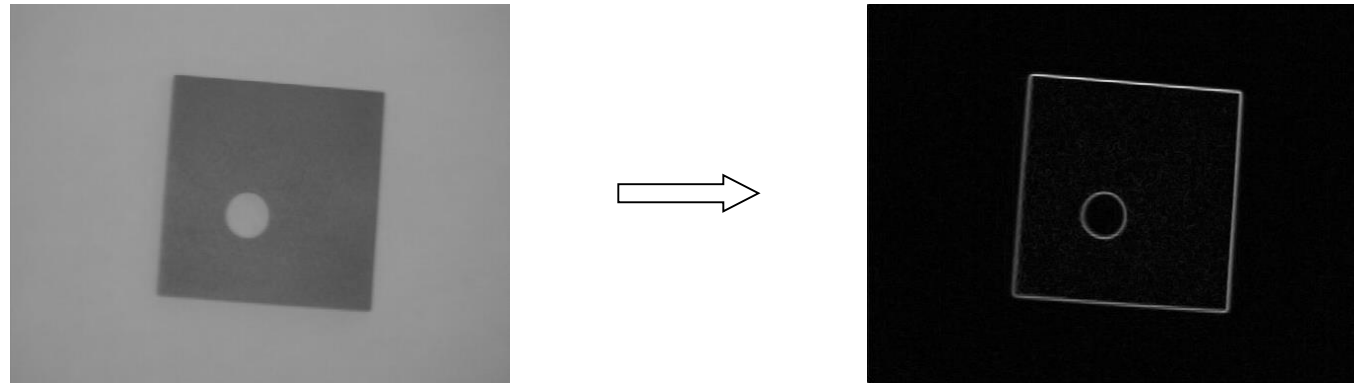
PREDAVATELJ: PATRICIO BULIĆ

ASISTENTA: DAVOR SLUGA, ROK ČEŠNOVAR

# Vaja 7

---

## Razpoznavanje robov v slikah



Pomembna tehnika na področju računalniškega vida

Sobelov operator

- Računanje gradientov intenzitete
- Kjer je gradient največji je verjetno rob

# Sobelov operator

---

$s(i,j)$  podaja intenziteto izvorne slike v točki  $(i,j)$

Iz intenzitete izbrane točke in njenih osmih sosed izračunamo intenziteto točke na novi sliki  $d(i,j)$ :

$$G_x = -s(i-1, j-1) - 2s(i-1, j) - s(i-1, j+1) + s(i+1, j-1) + 2s(i+1, j) + s(i+1, j+1) ,$$

$$G_y = -s(i-1, j-1) - 2s(i, j-1) - s(i+1, j-1) + s(i-1, j+1) + 2s(i, j+1) + s(i+1, j+1) ,$$

$$d(i,j) = \sqrt{G_x^2 + G_y^2} .$$

Intenzitete zunaj slike so 0

Največja dovoljena intenziteta je 255

	$i-1,j-1$	$i,j-1$	$i,j+1$		
	$i-1,j$	$i,j$	$i+1,j$		
	$i-1,j+1$	$i,j+1$	$i+1,j+1$		

# Vaja 7

---

Na učilnici se nahaja sekvenčna koda razpoznavanje robov.

Algoritem implementirajte na GPE s pomočjo knjižnice OpenCL.

Algoritem naj omogoča procesiranje slik poljubnih velikosti!

Uporabite deljeni pomnilnik, v katerega si naložite del slike, ki ga potrebujejo niti znotraj delovne skupine.

# Vaja 7

---

Preizkusite vaš algoritem pri generiranju slik različnih dimenzij

- 640x480, 800x600, 1600x900, 1920x1080, 3840x2160

Izmerite čas izvajanja in izračunajte pohitritev v primerjavi s serijskim algoritmom, ki teče na CPE

- Pohitritev  $S = T_{cpe} / T_{gpe}$
- Rezultate podajte kot komentar v kodi

Pri merjenju časa na GPE upoštevajte prenos vhoda slike na GPE, računanje in prenos izhodne v pomnilnik računalnika.

# Vaja 7

---

## Prevajanje in zaganjanje na NSC

- module load CUDA
- `gcc sobel.c -O2 -lm -lOpenCL -Wl,-rpath,./ -L./ -l:"libfreeimage.so.3" -o sobel`
- Zagon na GPE: `srun -n1 -G1 --reservation=fri sobel`

# Vaja 7

---

Rok za oddajo: 9. 12. 2021; zadnji možni: 16. 12. 2021.