

# Implementacija OpenCV biblioteka na Android platformi

---

**Brkić, Ivan**

**Master's thesis / Diplomski rad**

**2015**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:258097>

*Rights / Prava:* [In copyright](#)

*Download date / Datum preuzimanja:* **2022-01-25**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



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

**Sveučilišni studij**

**IMPLEMENTACIJA OPENCV BIBLIOTEKA NA  
ANDROID PLATFORMI**

**Diplomski rad**

**Ivan Brkić**

**Osijek, 2015.**

# SADRŽAJ

1. UVOD.....	1
2. ANDROID OPERACIJSKI SUSTAV .....	2
2.1. Android SDK.....	2
2.2. Kratka povijest Android operacijskog sustava.....	4
2.3. Arhitektura Android operacijskog sustava .....	5
2.4. Osnovna svojstva android aplikacija .....	7
2.4.1. Komponente.....	8
2.4.2. Korisničko sučelje .....	10
2.4.3. Resursi .....	11
2.4.4. Android manifest.....	12
3. OPENCV .....	14
3.1. Ugrađeni moduli.....	14
3.2. Detekcija oblika .....	15
3.3. Canny detektor rubova.....	16
3.4. Hough-ova transformacija za kružnice .....	18
3.4.1. Poznat polumjer r .....	18
3.4.2. Nepoznat polumjer r .....	19
4. POVEZIVANJE ANDROID OS-a I OPENCV BIBLIOTEKE .....	21
4.1. Instalacija OpenCV biblioteke.....	21
4.2. Pozivanje OpenCV metode iz android aplikacije .....	23
5. APLIKACIJA ZA DETEKCIJU KRUŽNICA.....	25
5.1. Opis izrađene aplikacije.....	25
5.2. Pohrana podataka .....	27
5.3. Korisničko sučelje .....	28
5.4. Obrada slika .....	29
5.5. Resursi i manifest datoteka .....	32
5.6. Eksperimentalni rezultati .....	34
6. ZAKLJUČAK.....	38
LITERATURA.....	39
POPIS UPOTRIJEBLJENIH KRATICA.....	40
SAŽETAK.....	41
ŽIVOTOPIS.....	42
PRILOZI .....	43

## SAŽETAK

U sklopu diplomskog rada uspješno je realizirana implementacija OpenCV biblioteke na Android platformu. Implementacija se zasniva na tri osnovna koraka: instalaciji CDT i NDK proširenja, preuzimanju OpenCV SDK i povezivanju korisničke aplikacije s OpenCV Manager-om.

OpenCV biblioteka detekciju oblika provodi u dva koraka. Prvi korak je određivanje rubova na slici korištenjem Canny metode, drugi korak je korištenje Hough transformacije (HT). U ovom radu su korištene HT za kružnice koje su definirane pomoću tri parametra: koordinate središta  $(a, b)$  i polumjer  $- r$ . Parametri kružnica se određuju preslikavanjem piksela iz geometrijskog u parametarski prostor.

Izrađena aplikacija omogućuje detekciju kružnica u stvarnom vremenu ili naknadnu detekciju kružnica na "mirnim" slikama. Također, moguće je promijeniti rezoluciju i parametre za detekciju kružnica u različitim uvjetima.

**Ključne riječi:** opencv, android, računalni vid, kružnice, hough transformacije, canny detektor rubova

As a part of this thesis, OpenCV library was successfully implemented into Android platform. Implementation is based on three main parts: installation of NDK and CDT extensions, download of OpenCV SDK and binding user's application to OpenCV Manager.

OpenCV's object detection is split in two parts. First part is use of Canny edge detection in order to detect edges in image, second part is use of Hough transform (HT). In this thesis HT for circles was used, which is defined by three parameters: center  $(a, b)$  and radius  $- r$ . Parameters are calculated by translating every pixel of an image into parameter space.

Application enables real-time circle detection. It also enables later circle detection on still images. Also, it is possible to change image resolution and circle detection parameters for use in various image conditions.

**Title:** Implementation of OpenCV libraries on Android platform

**Keywords:** opencv, android, computer vision, circles, hough transform, canny edge detection