

Technická správa

# Program pre zobrazovanie vlastností obrazu

23. marca 2017

Autor: Tibor Mikita, [xmikit01@stud.fit.vutbr.cz](mailto:xmikit01@stud.fit.vutbr.cz)  
Fakulta Informačních Technologí  
Vysoké Učení Technické v Brně

# Obsah

<b>1</b>	<b>Zloženie tímu</b>	<b>1</b>
<b>2</b>	<b>Zadanie</b>	<b>1</b>
<b>3</b>	<b>Náš cieľ</b>	<b>1</b>
<b>4</b>	<b>Štúdium problematiky</b>	<b>1</b>
4.1	Dostupné technológie . . . . .	1
4.2	Informácie, ktoré je možné získať . . . . .	2
4.3	Podporované formáty obrázkov . . . . .	2
4.4	Testovacie dáta . . . . .	2
<b>5</b>	<b>Použité technológie</b>	<b>3</b>
<b>6</b>	<b>Rozdelenie práce v tíme</b>	<b>3</b>
<b>7</b>	<b>Aktuálny stav</b>	<b>3</b>
7.1	Screenshots z aplikácie . . . . .	4
7.2	Problémy, na ktoré sme stihli naraziť . . . . .	5

# 1 Zloženie tímu

- Tibor Mikita, [xmikit01@stud.fit.vutbr.cz](mailto:xmikit01@stud.fit.vutbr.cz)
- Martin Matejčík, [xmatej46@stud.fit.vutbr.cz](mailto:xmatej46@stud.fit.vutbr.cz)

# 2 Zadanie

Zadaním je vypracovať program, ktorý bude schopný zobrazovať rôzne údaje o načítanom obrázku.

# 3 Náš cieľ

Výsledkom tohto projektu bude aplikácia s grafickým užívateľským rozhraním, do ktorej bude možné nahráť ľubovoľný obrázok. Po načítaní obrázka do aplikácie sa, v užívateľsky prívetivom prostredí, zobrazia rôzne informácie. Budú to informácie týkajúce sa súboru ako takého, ale aj informácie súvisiace so samotným obrázkom, prípadne ďalšie informácie, ako je napríklad informácia o pixeli v mieste kurzora myši, histogram obrázka a iné.

# 4 Štúdium problematiky

## 4.1 Dostupné technológie

Rozhodli sme sa navrhnutú aplikáciu implementovať v jazyku Java. Existuje niekoľko spôsobov ako v Jave pracovať s OpenCV knižnicou. Najvhodnejšími kandidátmi sú:

- JavaCV
  - + rozhranie pre OpenCV napísané v Jave
  - + zahrňuje množstvo knižníc z oblasti počítačového videnia
  - odlišné názvy funkcií od OpenCV
  - horšia dokumentácia
- OpenCV packaged by OpenPnP
  - + práca s natívnou knižnicou OpenCV priamo v Jave (Java bindings)
  - + chválitebná dokumentácia a množstvo príkladov
  - „len“ OpenCV funkcie

Pre získanie MIME typu súboru v Jave, bez ohľadu na príponu, je možné využiť niekoľko metód. Ako najspoľahlivejšie sa zdajú byť tieto:

- jMimeMagic
  - + spoľahlivé
  - + jednoduché použitie

- otravné „warning“ oznámenia pri spustení
- nepodporuje TIFF formát
- Apache Tika
  - + mocný nástroj
  - + dobrá podpora
  - + vie to rozoznať tisíce rôznych formátov(aj TIFF!)

## 4.2 Informácie, ktoré je možné získať

- o súbore
  - názov
  - MIME typ
  - veľkosť
- o obrázku
  - šírka a výška v pixeloch
  - bitová hĺbka
  - počet kanálov
  - počet unikátnych farieb
- o pixeli
  - pozícia v obrázku
  - farba, resp. intenzita
- rôzne
  - histogram
  - rozdelenie obrázka na jednotlivé kanály

## 4.3 Podporované formáty obrázkov

Navrhnutá aplikácia by mala podporovať tieto formáty obrázkov: BMP, JPG, PNG, TIFF. Dôležité zistenie je, že knižnica OpenCV nevie pracovať s GIF, tým pádom ani naša aplikácia to nebude podporovať.

## 4.4 Testovacie dáta

Testovacie dáta budú prirodzene pozostávať z rôznych obrázkov rôzneho formátu(JPG, PNG, BMP) a veľkosti. Obrázky budú mať rôzny počet kanálov(RGB, RGBA, GrayScale). Súčasťou testovacej sady budú napríklad aj súbory bez prípony, alebo súbory s nesprávnou príponou, aby bolo možné otestovať MIME typ obrázka nezávisle na prípone v názve súboru. Testovacie dáta budú stiahnuté z nejakej voľne prístupnej databázy.

## 5 Použité technológie

- Java 8
- JavaFX - GUI
- OpenCV 3.2.0 (packaged by OpenPnP) - práca s obrázkami
- Apache Tika Core - MIME typ
- JFoenix - material design
- Gradle - build system

## 6 Rozdelenie práce v tíme

Martin:

- základná štruktúra aplikácie podľa MVC modelu
- prepojenie použitých technológií
- rozdelenie obrázku na jednotlivé kanály a ich vzájomné zobrazovanie

Tibor:

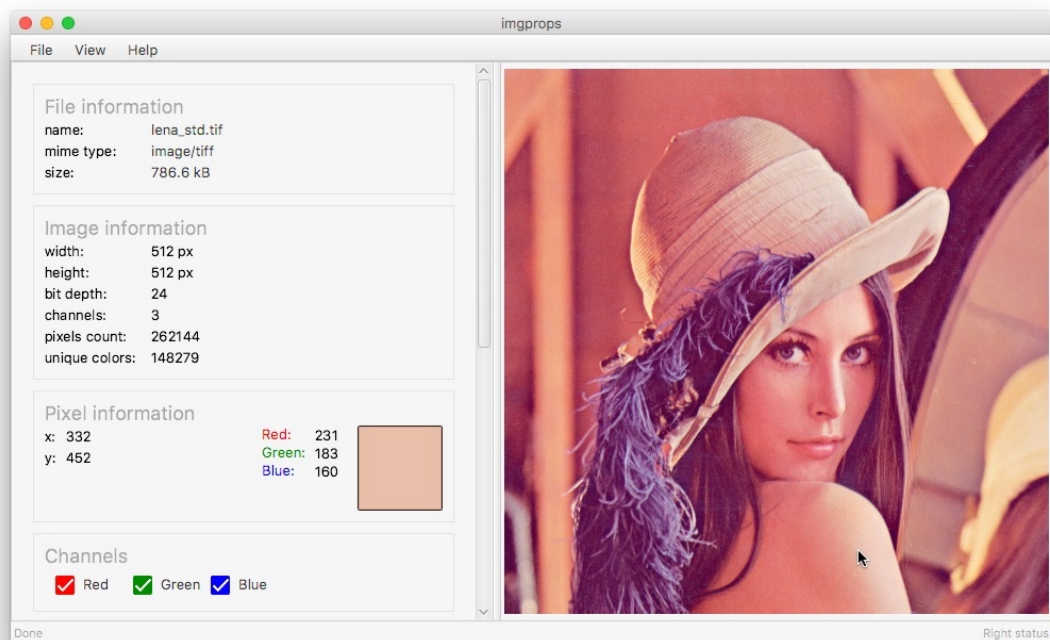
- načítanie obrázku zo súboru a získanie základných informácií o obrázku a o súbore ako takom
- histogram
- zobrazovanie informácií o pixeli pod kurzorom myši
- zhromaždenie testovacej sady obrázkov

## 7 Aktuálny stav

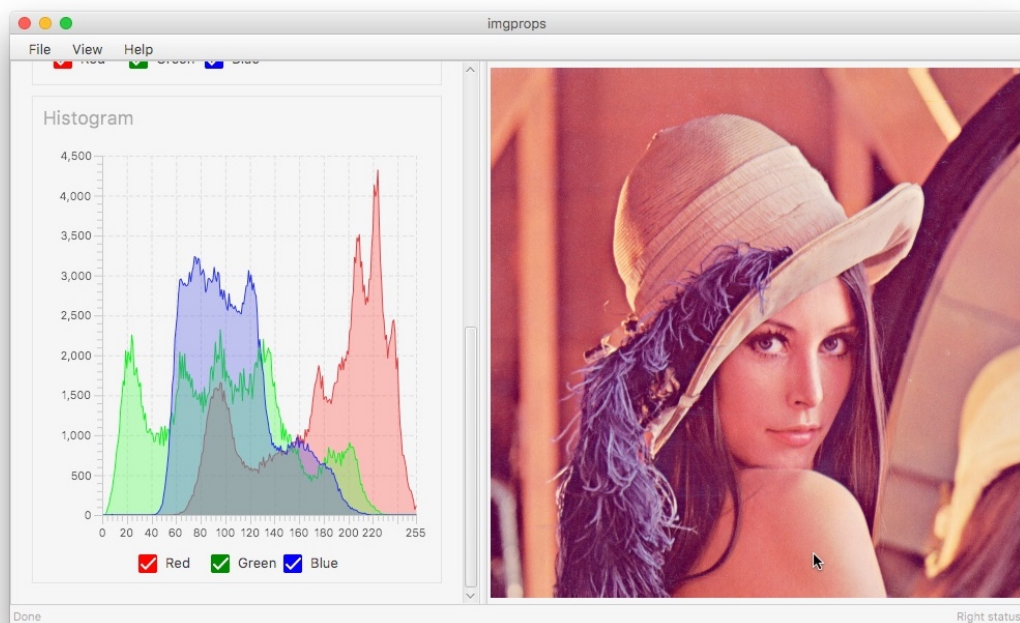
Toto sme už stihli implementovať:

- zobrazovanie informácií
  - o súbore(názov, MIME typ, veľkosť v bytoch)
  - o obrázku(šírka a výška v pixeloch, bitová hĺbka, počet kanálov, celkový počet pixelov)
  - o pixeli v mieste kurzora myši(x-ová a y-ová súradnica, farba v RGB, resp. intenzita)
- histogram rozdelený podľa RGB zložiek(jednotlivé zložky je možné schovávať a znova zobrazovať)
- zobrazovanie a schovávanie jednotlivých farebných zložiek obrázku
- ďalšie možnosti prístupné cez menu
  - znovuotvorenie nedávno otvorených obrázkov
  - schovávanie a zobrazovanie jednotlivých sekcií s informáciami

## 7.1 Screenshoty z aplikácie



Obr. 1: Základné informácie



Obr. 2: Ukážka histogramu

## 7.2 Problémy, na ktoré sme stihli naraziť

Funkcia `imread()` s príznakom `IMREAD_UNCHANGED` zhodí celý program, ak vstupom je 16-bitový `tiff` súbor. Preto sme sa rozhodli vymeniť príznak `IMREAD_UNCHANGED` za dvojicu príznakov `CV_LOAD_IMAGE_ANYCOLOR` a `CV_LOAD_IMAGE_ANYDEPTH`. Tým ale strácame informáciu o pôvodnom počte kanálov, pretože OpenCV pôvodný obrázok prekonvertuje do greyscale alebo RGB modelu a teda vždy pracujeme s jedným alebo tromi 8-bitovými kanálmi. Pôvodná informácia o počte kanálov a bitovej hĺbke pre určité obrázky (napr. spomínaný 16-bitový `tiff`) je teda nedostupná.