|  |  |  |
| --- | --- | --- |
|  | | |
| **Závěrečná studijní práce**  **dokumentace** | | |
| **Rozpoznávání registračních značek automobilů pomocí OpenCV a ukládaní dat do DB aplikace** | | |
| Dominik Stuchlý | | |
|  | | |
|  | |  |
| **Obor:** | 18-20-M/01 INFORMAČNÍ TECHNOLOGIE  se zaměřením na počítačové sítě a programování | |
| **Třída:**  **Školní rok:** | IT4  2016/2017 | |

#### Poděkování

*Chtěl bych velice poděkovat vedoucímu mé práce Ing. Petru Grussmanovi za navrhnutí téhle problematiky, kterou jsem v práci řešil, a která mě velice zaujala. Také bych chtěl poděkovat za dobré vedení, postup a nové nápady, jak v projektu pokračovat a jak vyřešit problémy, které jsem sám nedokázal zvládnout. A také bych rád poděkoval svým spolužáků, se kterýma jsem pracoval na podobném zadání, za poznatky a nápady, se kterými se se mnou podělili.*

Prohlašuji, že jsem závěrečnou práci vypracoval samostatně a uvedl veškeré použité   
informační zdroje.

Souhlasím, aby tato studijní práce byla použita k výukovým účelům na Střední průmyslové   
a umělecké škole v Opavě, Praskova 399/8.

V Opavě 31. 12. 2016

*podpis autora práce*

**ANOTACE**

Projekt se zabývá prací s knihovnou OpenCV a její konfigurací, aplikací a použitím v praxi. Úkolem je získat z video streamu data, jako jsou registrační značky aut, jaké zemi patří tato značka a s jakou přesností byla získána. Pomocí aplikace IP Webcam na mobilním telefonu, bylo streamovano video jedoucích aut a následně na počítači spojeno s knihovnou OpenALPR, která uloží obrázky aut a získá z nich data. Knihovna také odešle data na určitou adresu. Ty data následně pomocí aplikace napsané v Pythonu jsou zpracovány, uloženy do databázové aplikace a s nimi jsou provedeny operace.

**Klíčová slova**

OpenCV, rozpoznávání SPZ, databáze, statistika, auta, python, videokamera

OBSAH

[Poděkování 2](#_Toc502436122)

[Úvod 6](#_Toc502436123)

[1 Seznámení se s knihovnami a registračními značkami 7](#_Toc502436124)

[1.1 Knihovna OpenCV 7](#_Toc502436125)

[1.2 OpenALPR 8](#_Toc502436126)

[1.3 Registrační značky 9](#_Toc502436127)

[2 Využité technologie 10](#_Toc502436128)

[2.1.1 OpenCV 10](#_Toc502436129)

[2.1.2 IP Webcam 10](#_Toc502436130)

[2.1.3 Python 10](#_Toc502436131)

[2.1.4 C++ 10](#_Toc502436132)

[2.1.5 Lenovo S60-a 10](#_Toc502436133)

[3 Způsoby řešení a použité postupy 11](#_Toc502436134)

[3.1 Instalace OpenCV na Linux 11](#_Toc502436135)

[3.2 Instalace OpenALPR 12](#_Toc502436136)

[3.3 Test funkčnosti knihovny OpenALPR 13](#_Toc502436137)

[3.4 Vytvoření video streamu 14](#_Toc502436138)

[3.5 Konfigurace OpenALPR 15](#_Toc502436139)

[3.6 DB aplikace 17](#_Toc502436140)

[Závěr 19](#_Toc502436141)

[Seznam použitýCH INFORMAČNÍCH ZDROJů 20](#_Toc502436142)

Úvod

OpenCV zajišťuje velkou škálu možností, jak pracovat v reálném čase s obrázky či videi pořízených za denních podmínek a získat z nich informace. Představa, že by člověk stál na dálnici a rychle četl registrační značky jedoucích vozidel a měl si je všechny zapamatovat je absurdní, proto se mi líbila možnost tuhle problematiku sám zautomatizovat a získané značky uložit do databáze a provést s nimi různé statistické operace. Podobné mobilní systémy používá policie zejména v USA, ti je mají na vozidlech nainstalované a při střetu s podezřelými vozidly mohou snímat jeho poznávací značku a tu porovnávat ve svých databázích. To jim pomáhá ve vyšetřování.

Mým cílem bylo tedy správně nainstalovat a nakonfigurovat knihovnu OpenCV a zjistit, jak nejlépe zpracovat obrázky či video automobilů a z nich získat registrační značku. Poté získané data zpracovat ve vlastní aplikaci a uložit je do databáze kde s nimi bude možné provádět různé operace, například jestli už auto s danou poznávací značkou projelo někdy tímto místem a kolikrát.

Streamování videa jsem vyřešil stáhnutím aplikace IP webcam na můj chytrý telefon. IP webcam živě vysílá video na určitou IP adresu na kterou jsem se mohl poté na svém počítači připojit a video sledovat.

Nejvhodnější knihovna k použití byla OpenALPR, která je udělaná konkrétně pro rozpoznávání registračních značek automobilů. V konfiguračních souborech lze snadno propojit s video streame a určit kam bude získané data posílat, ty pak zpracuje moje aplikace napsaná pythonu a uloží je do databáze.

# Seznámení se s knihovnami a registračními značkami

## Knihovna OpenCV

OpenCV (Open Source Computer Vision Libary) je open source softwarová knihovna určená pro práci s počítačovým viděním. V roce 1999 ji začala vyvíjet společnost Intel. Nyní je pod licencí BSD, což ji dělá jednoduché v podnikání, při využívání a jakékoliv smysluplné upravování a vylepšování je vítáno.

Knihovna má více než 2500 optimalizovaných algoritmů, které zahrnují komplexní sadu klasických a nejmodernějších počítačové vidění a algoritmy pro strojové učení. Tyto algoritmy jsou používané k detekci a rozpoznání obličejů, identifikaci objektů, klasifikaci lidského jednání ve videích, sledování pohyblivých objektů, extrahování 3D modelů objektů, hledání podobných obrázků z databáze obrázků, odstranění červených očí ze snímků pořízených s bleskem atd.

OpenCV má přes 47 tisíc uživatelů v komunitách a odhadovaný počet stažení přesahuje 14 miliónů. Knihovnu široce využívají společnosti jako je Google, Yahoo, Microsoft, Intel, Sony, Honda a Toyota, ale také v jiných firmách, výzkumných organizacích a vládních orgánech. Využití nalezneme třeba v robotice, bezpečnostních systémech, stereo-vizi, lékařských diagnostikách a ve spoustě dalších odvětvích. Konkrétní případy jsou například, scelování obrazů dohromady ve Streetview, sledování důlních zařízení v Číně, pomoc robotům při navigaci manipulaci s předměty, odhalování topících se lidí v plaveckých bazénech, kontrola značek na produktech v továrnách a další.

## OpenALPR

OpenALPR (Automatic License Plate Recognition library) je open source knihovna určená pro rozpoznávání registračních značek vozidel. Napsaná je v C++ s vazbami na C #, Javu, Nodes.js a Python. Knihovna analyzuje obrázky a video stream a z nich identifikuje registrační značky. Výstupem je jejich textové zobrazení.

Software může být využit mnoha různými způsoby. Například:

1. Rozpozná registrační značky z kamerových streamů. Výsledky jsou přehledné, vyhledatelné a můžou spustit upozornění. Uložiště dat může být v cloudu nebo můžou být ukládány ve
2. Rozpozná registrační značky z kamerových streamů a výsledky pošle do vaší vlastní aplikace.
3. Zpracuje video soubor a uloží jeho značku do CSV a dá do SQLite databáze.
4. Analyzuje statické obrázky z příkazového řádku.
5. Vloží rozpoznané poznávací značky do vaší aplikace přímo v kódu (C, C++, C#, VB.NET, Java, Python, Node.js)

## Registrační značky

Registrační značka (nebo také SPZ – státní poznávací značka) je ze zákona povinný prvek motorových vozidel. Obsahuje alfanumerické označení, toto označení je unikátní pro každé vozidlo. Značky různých zemí se můžou velice lišit, velikostí, barevně nebo i pořadím čísel a písmen.

Standartní značky v ČR:

* cz #@#####
* cz #@@####

(#=číslo 0-9, @ =písmena bez diakritiky a s výjimkou G, O, Q, W)



*obr. 1 česká SPZ*

Jen na ukázku, jak mohou vypadat značky v různých zemích.

Švýcarsko: USA Louisiana:



*obr. 3 USA SPZ*

*obr. 2 švýcarská SPZ*

Izrael: Rusko:

*obr. 4 izraelská SPZ*

*obr. 5 ruská SPZ*



# Využité technologie

## OpenCV

OpenCV (Open Source Computer Vision Library) je open source multiplatformní knihovna. Určená je především pro počítačové vidění a zpracování obrazu v reálném čase. Knihovna má rozhraní v C, C++, Pythonu, Javě a je podporována pro Windows, Linux, iOS a Android.

## IP Webcam

IP Webcam je freeware aplikace pro android, která změní telefon na síťovou kameru s mnoha možnostmi zobrazení. Může zobrazit fotoaparát na libovolné platformě pomocí přehrávače VLC nebo webového prohlížeče tím, že streamuje video uvnitř WiFi sítě.

## Python

Python je vysokoúrovňový skriptovací jazyk. Nabízí dynamickou kontrolu datových typů a také podporuje různá programovací paradigmata, včetně objektově orientovaného, imperativního, procedurálního nebo funkcionálního. Je vyvíjen jako open source. Zdarma nabízí instalační balíky pro Unix, Windows, iOS.

## C++

C++ je multiparadigmatický programovací jazyk, který patří mezi nejrozšířenější programovací jazyky současnosti. Je rozšířením jazyka C. Podporuje několik programovacích paradigmat jako je objektově orientované programování a generické programování.

## Lenovo S60-a

Můj chytrý telefon Lenovo S60 jsem použil, abych na něho mohl nainstalovat aplikaci IP Webcam. Pomocí 13 Mpix fotoaparátu jsem nahrával video, které jsem následně přes WiFi streamoval.

# Způsoby řešení a použité postupy

## Instalace OpenCV na Linux

Požadované balíčky:

* GCC 4.4.x nebo novější
* CMake 2.6 nebo vyšší
* Git
* GTK+2.x nebo vyšší, včetně záhlaví (libgtk2.0-dev)
* pkg-config
* Python 2.6 nebo novější a Numpy 1.5 nebo novější s vývojářskými balíčky (python-dev, python-numpy)
* ffmpeg nebo libav vývojové balíčky: libavcodec-dev, libavformat-dev, libswscale-dev
* [volitelně] libtbb2 libtbb-dev
* [volitelně] libdc1394 2.x
* [volitelně] libjpeg-dev, libpng-dev, libtiff-dev, libjasper-dev, libdc1394-22-dev

Balíčky lze nainstalovat pomocí terminálu a následných příkazů nebo pomocí programu Synaptic Manager:

[compiler] sudo apt-get install build-essential

[required] sudo apt-get install cmake git libgtk2.0-dev pkg-config libavcodec-dev libavformat-dev libswscale-dev

[optional] sudo apt-get install python-dev python-numpy libtbb2 libtbb-dev libjpeg-dev libpng-dev libtiff-dev libjasper-dev libdc1394-22-dev

## Instalace OpenALPR

//Stáhnutí a instalace deamona:

sudo apt-get update && sudo apt-get install -y openalpr openalpr-daemon openalpr-utils libopenalpr-dev

//Když používáme deamona je nutné nainstalovat beanstalkd:

sudo apt-get install beanstalkd

//Získání nejnovější verze kódu z GitHubu:

git clone <https://github.com/openalpr/openalpr.git>

//Vytvoření složky pro kompilaci:

cd openalpr/src

mkdir build

cd build

//Vytvoření prostředí kompilace:

cmake -DCMAKE\_INSTALL\_PREFIX:PATH=/usr -DCMAKE\_INSTALL\_SYSCONFDIR:PATH=/etc ..

//Kompilace knihovny:

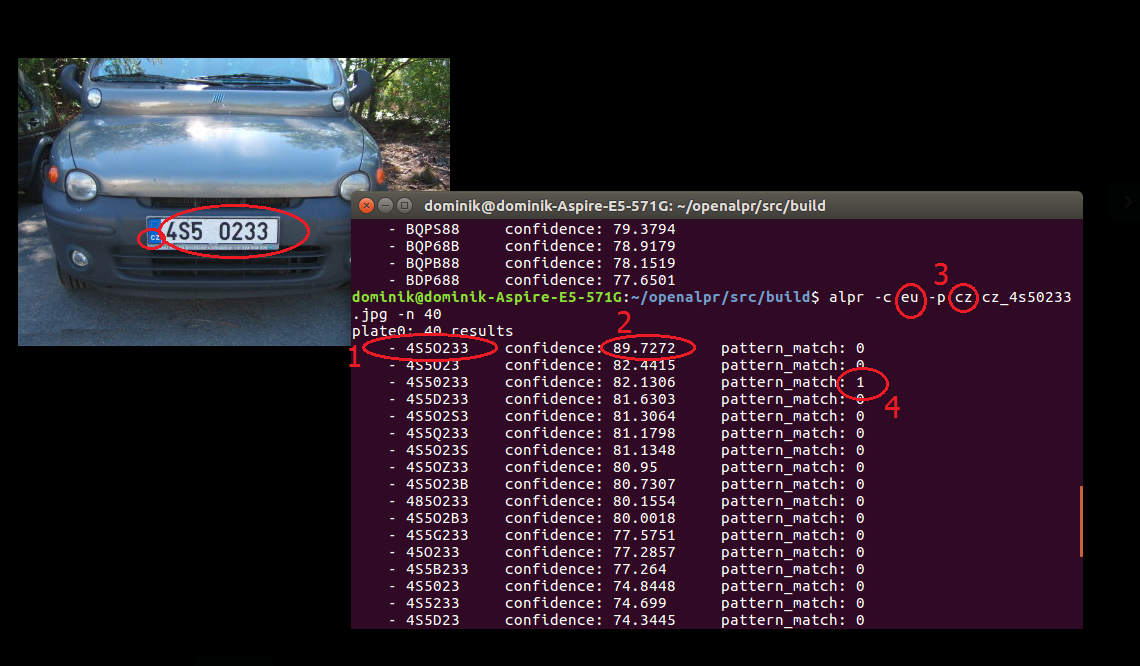
make

//Instalace binárních souborů do místního systému:

sudo make install

* popis řešení úkolu včetně, použité postupy a jejich vysvětlení, způsoby testování funkčnosti, parametry výrobku (programu, hotového řešení), schémata, obrázky z tvorby a finálního provedení, výpočty, použité příkazy…

## Test funkčnosti knihovny OpenALPR



*obr. 6 test funkčnosti*

Příkaz pro získání čísla registrační značky ze statického obrázku:

~/openalpr/src/build$ alpr -c eu -p cz cz\_4s50233.jpg -n 40

Dostaneme 40 výsledků, jak je na konci příkazu zadáno. Na obrázku vidíme (1), že se výsledné číslo “4S50233“ poznávací značky shoduje s jistotou 89,7 % (2) s obrázkem auta. V příkaze jsem také nastavil (3), aby knihovna značku porovnala s druhem značek Evropské unie a konkrétně s českou poznávací značkou a vidíme, že se shoda našla (4).

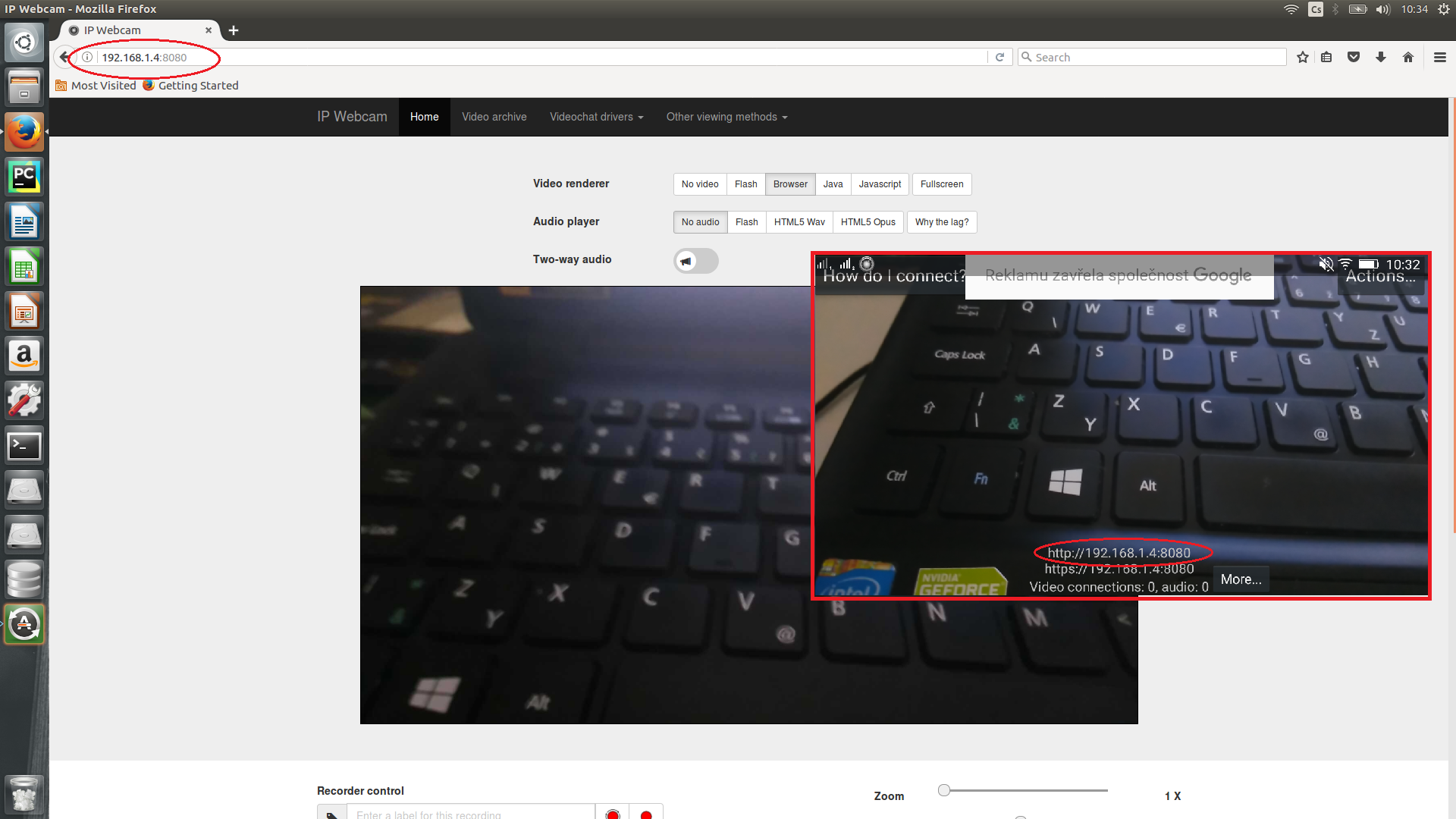
## Vytvoření video streamu

Na můj mobil Lenovo S60 jsem nainstaloval z Obchod Play aplikaci IP Webcam, která zajišťuje stream videa.

Odkaz na aplikaci:

<https://play.google.com/store/apps/details?id=com.pas.webcam&hl=cs>

Mobil a počítač musí být připojené na stejné síti, protože IP webcam vytvoří server v lokální síti. Po zapnutí aplikace a stisknutí “Start server“ se spustí video stream na níže uvedené adrese. Na tu adresu se poté můžeme na počítači pomocí webového prohlížeče připojit a video si zobrazit. Kameru lze ovládat jak na mobilu, tak také na počítači.



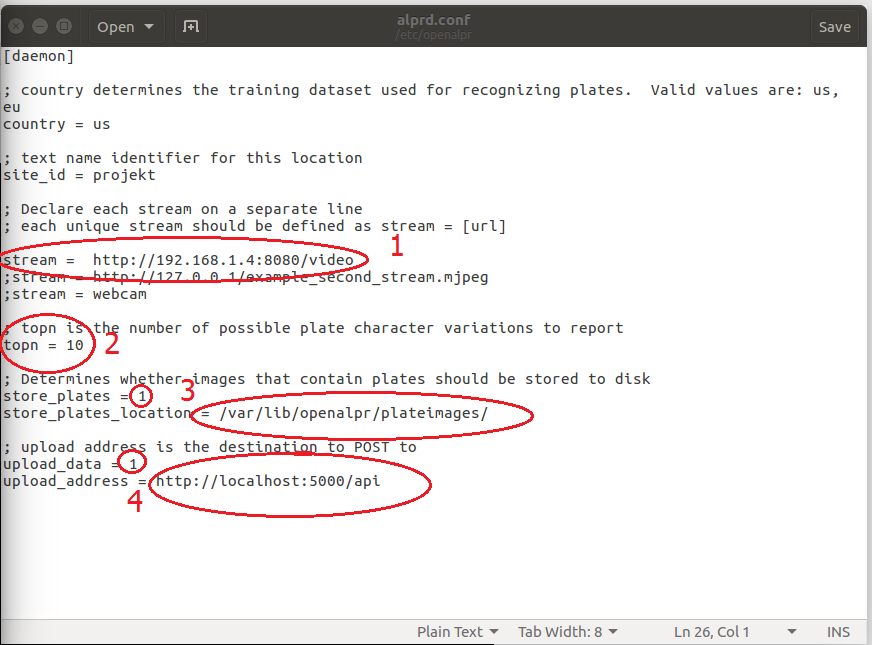
*obr. 7 video stream*

## Konfigurace OpenALPR

Nejpodstatnější bylo nastavování knihovny v souboru alprd.conf, který se nachází v adresáři /etc/openalpr:

1. Odkaz na video stream, který bude OpenALPR zpracovávat a získávat z něho poznávací značky aut.
2. Počet možných variací značek, který program nahlásí.
3. Povolení pro to, aby program ukládal obrázky, na kterých rozpozná registrační značku auta z videa. Cestu kam obrázky ukládat si můžeme vybrat, nebo nechat tu v předvolbách /var/lib/openalpr/plateimages/.
4. Velice důležité je také nastavit kam získané data bude OpenALPR posílat. Já data posílal do lokální sítě na <http://localhost:5000/api> a odtud pak v aplikaci napsané v pythonu data zpracovával.

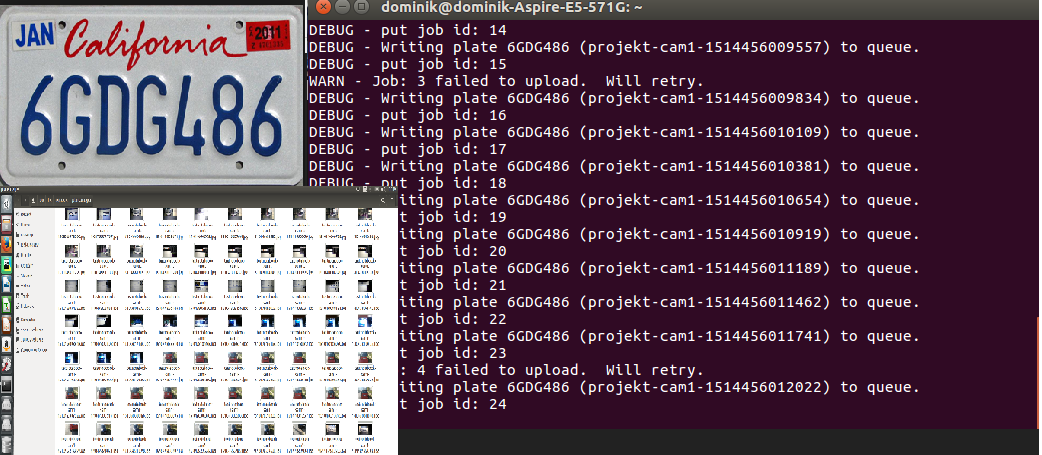
*obr. 8 konfigurační soubor*



Pomoci tohoto příkazu můžeme sledovat protokoly OpenALPR. Jestli všechno probíhá v pořádku, spojení s kamerou, získání dat, uložení obrázku a nahraní dat na <http://localhost:5000/api>.

tail -f /var/log/alpr.log

*obr. 9 sledování protokolů*

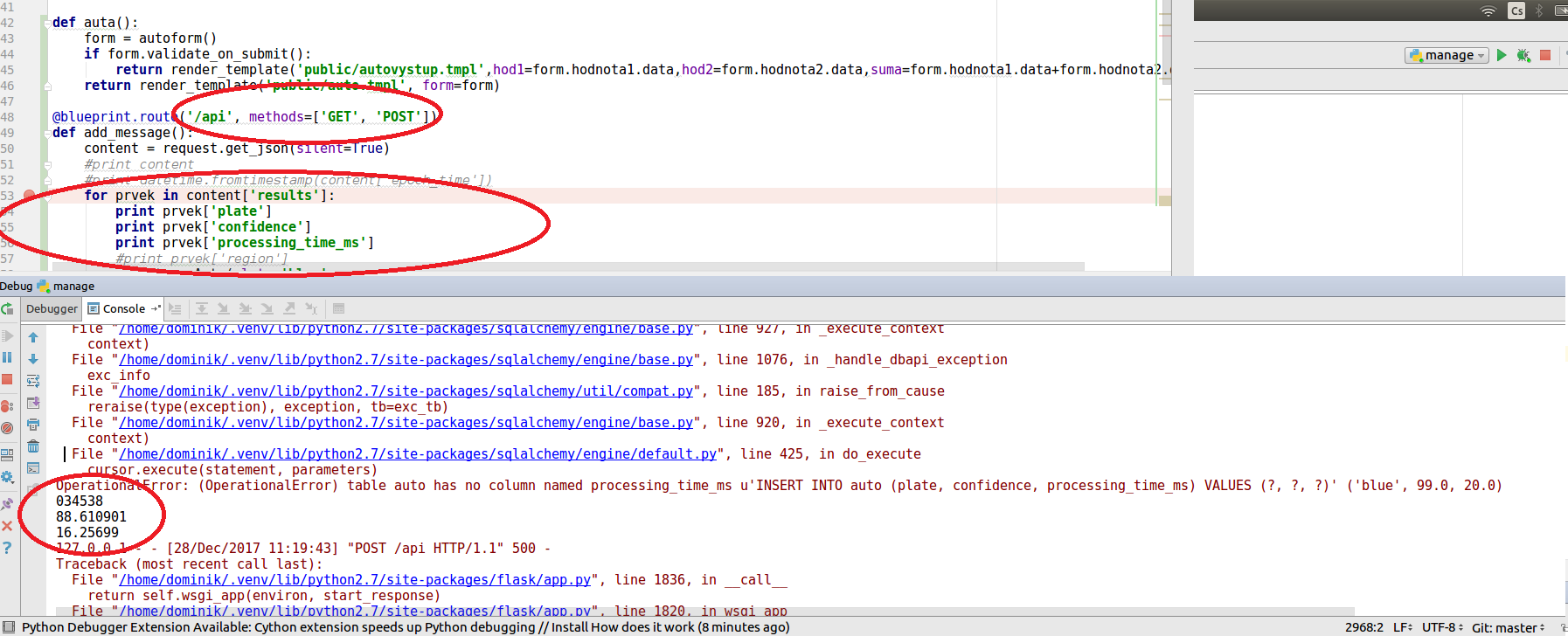


Po zaměření kamery na poznávací značku, uloží OpenALPR obrázek registrační značky do /var/lib/openalpr/plateimages/.

## DB aplikace

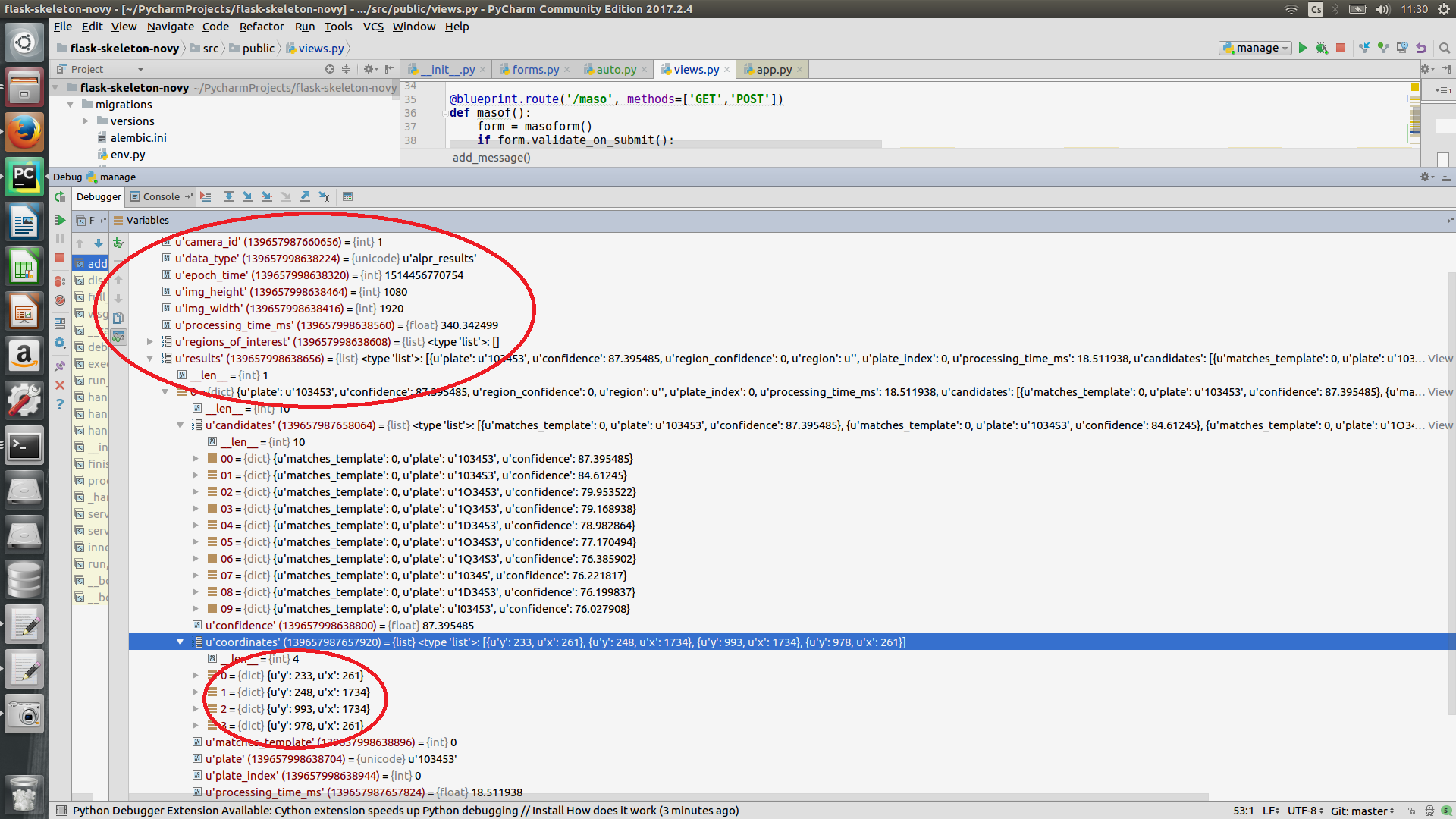
Zde můžeme vidět, že databázová aplikace je zatím schopna nám vypsat poznávací značku, jistotu přesnosti, a čas, jak dlouho operace trvala.

V prvním kroužku je napsané, že data bude brát z <http://localhost:5000/api>, to je adresa, kterou jsem v souboru alprd.conf nastavil jako adresu pro to, kam mají mířit odesílaná data. V druhém kroužku můžeme vidět, jak jednoduše výsledky vypsat a ve třetím je již vidíme vypsané.



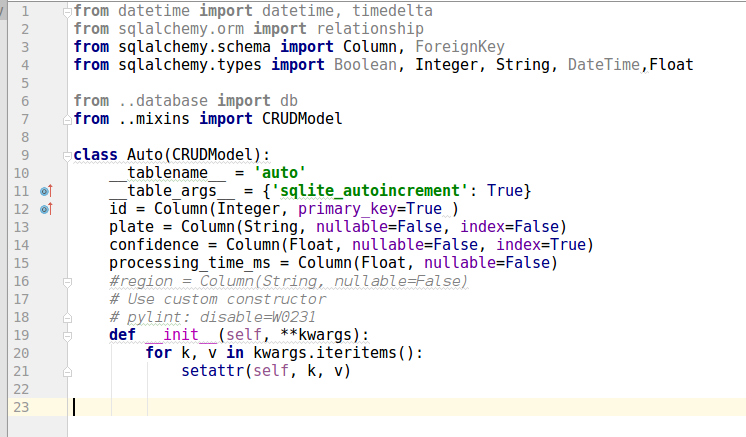
*obr. 10 zobrazení dat*

Tady si můžeme zobrazit všechna data, která OpenALPR pošle. Kód registrační značky, identifikační číslo kamery, řekne nám, že to jsou výsledky přicházející z od alpr, jak dlouho proces trval, parametry obrázky (výška, šířka), nebo také přesné souřadnice na obrázku, ve kterých se poznávací značka nachází



*obr. 11 poslaná data*

Kód na vytvoření prozatimní jednoduché databáze, kam se uloží získané data:



*obr. 12 databáze*

# Závěr

Při práci na projektu jsem se seznámil s knihovnami OpenCV a OpenALPR. Tyto knihovny se mi podařilo stáhnout a nainstalovat. Důležitým bodem bylo propojení video streamu, vytvořeném aplikací IP webcam, s OpenALPR, zde byla klíčová práce se souborem alprd.conf. Nastavením v konfiguračním souboru jsem také zajistil, aby čísla poznávacích značek, které OpenALPR získá z videa poslal na adresu, ze které je pak získá a zpracuje jednoduchá aplikace. Ta pak značky uložila do databáze.

Na projektu bych chtěl ještě vylepšit, abych využil obrázky registračních značek aut, pořízených z videa knihovnou OpenALPR a například je nahrál také do databáze a spojil je s určitou značkou. OpenALPR dokáže porovnat také typ poznávací značky, z jaké je země (např. eu, us, gb), ale zatím se mi nepodařilo, aby to určil stejně jako číslo značky automaticky a nahrál je do databáze společně. Kromě toho, bych chtěl do budoucna vylepšit mou databázovou aplikaci, aby měla více funkcí a také bych chtěl zapracovat na vytvořené databázi.

V praxi by se dala aplikace použít, kdybychom chtěli mít přehled nad konkrétním místem, kde projíždějí auta a my bychom zde umístili kameru a získávali jejich poznávací značky. V praxi již podobné aplikace fungují, zejména v bezpečnostních systémech, například rychlostní radary na silnicích, závorové systémy, pátrání po odcizených vozidlech.

Seznam použitýCH INFORMAČNÍCH ZDROJů

[] Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-12-29].

[2] OpenCV. *OpenCV: Open Source Computer Vision Library* [online]. OpenCV team [cit. 2017-12-29]. Dostupné z: https://opencv.org

[3] OpenALPR: openalpr documentation. *OpenALPR: openalpr documentation* [online]. OpenALPR Technology [cit. 2017-12-30].

Dostupné z: http://doc.openalpr.com/index.html

[4] *GitHub: GitHub OpenALP* [online]. Commerce, MI: matthill [cit. 2017-12-30]. Dostupné z: <https://github.com/openalpr/openalpr>