

TECHNICKÁ KOMUNIKÁCIA

AUTENTIFIKÁCIA OPERÁTORA NA ZÁKLADE
TVÁRE

1 Zdrojové kódy

Všetky kódy použité na implementáciu navrhovaného riešenia sú dostupné na:
<https://github.com/TheBranislav/AUTENTIFIKACIA-OPERATORA-NA-ZAKLADE-TVARE>

Váhy pre natrénovanú neurónku si používateľ môže stiahnuť na:
https://drive.google.com/drive/folders/1ODS3gx9kxsiGNujK2TMFNGD3U_WbfCw2

1.1.1 Štruktúra priečinka - Windows

```
projekt/  
├── faceRecognitionApp.py  
├── model_PAcc0.6471_NAcc0.9756.pth  
├── deploy.prototxt  
├── xiapi.py  
├── xidefs.py  
├── __init__.py  
├── res10_300x300_ssd_iter_140000.caffemodel  
└── operators/
```

1.1.2 Štruktúra priečinka - Linux

```
projekt/  
├── scripts/  
│   ├── _pycache_  
│   ├── operators/  
│   ├── deploy.prototxt  
│   ├── res10_300x300_ssd_iter_140000.caffemodel  
│   ├── recognition_publisher  
│   └── model_PAcc0.6471_NAcc0.9756.pth  
├── src/  
│   └── prediction.cpp  
├── CMakeLists  
└── package
```

2 Návod na inštaláciu riešenia

2.1 Návod na spustenie – Windows

Tento návod vás prevedie procesom spustenia aplikácie na rozpoznávanie operátora na operačnom systéme Windows. Je nutné stiahnuť priečinok `facial_recognition_windows`, ktorý tvorí jednu z príloh diplomovej práce. Tento priečinok obsahuje všetky potrebné súbory pre spustenie programu na OS Windows.

2.1.1 Požiadavky

- Windows 10 alebo 11
- GPU s podporou CUDA (je potrebné overiť verziu kompatibilnú s Vašou grafickou kartou)
- Python 3.8+ (Odporúča sa verzia 3.10)
- Inštalácia XIMEA kamera – xiAPI (<https://www.ximea.com/support/downloads>)

2.1.2 Vytvorenie virtuálneho prostredia (odporúčané)

Je odporúčané zadať tento kód do `.cmd`.

```
python -m venv venv  
venv\Scripts\activate
```

2.1.3 Inštalácia knižníc

Pre spustenie programu je potrebné stiahnuť nasledovné knižnice

```
pip install opencv-python  
pip install matplotlib  
pip install pillow  
pip install xiapi  
pip install numpy  
pip install torch torchvision --index-url  
https://download.pytorch.org/whl/cu118
```

Posledný príkaz je nutné upraviť podľa verzie grafickej karty pre CUDA, verzia 11.8. je kompatibilná s verziou python 3.10 a s grafickou kartou Nvidia 2060. (Odporúča sa pre tento krok využiť umelú inteligenciu prompt: „Mám grafickú kartu XXXX a python verzie 3.10,

ako mám upraviť tento príkaz „`pip install torch torchvision --index-url https://download.pytorch.org/whl/cu118`“, aby som vedel spustiť tento kód.“ (Odporúča sa vložiť kód)).

2.1.4 Spustenie aplikácie

Dvojklikom na súbor *faceRecognitionApp.py* alebo vložení tohto príkazu do .cmd *python faceRecognitionApp.py* kód spustíme. Po spustení:

- Aplikácia sa otvorí v GUI režime.
- Kamera sa aktivuje (XIMEA).
- Ak existujú tváre v ./operators, systém ich načíta a rozpoznáva v reálnom čase.
- na OS windows, publisher nepublikuje správy na žiadny node. Aplikácia je však plne funkčná, okrem tejto vlastnosti.

2.2 Návod na spustenie – Linux

Tento návod vás prevedie procesom spustenia aplikácie na rozpoznávanie operátora na operačnom systéme Linux. Je nutné stiahnuť a rozbaliť priečinok *facial_recognition*, ktorý tvorí jednu z príloh diplomovej práce. Tento priečinok obsahuje všetky potrebné súbory pre spustenie programu na OS Linux.

2.2.1 Požiadavky

- Ubuntu 22.04 LTS
- GPU s podporou CUDA (je potrebné overiť verziu kompatibilnú s Vašou grafickou kartou)
- Python 3.8+ (Odporúča sa verzia 3.10)
- ROS2 Humble (<https://docs.ros.org/en/humble/Installation.html>)
- Inštalácia XIMEA kamera – xiAPI (<https://www.ximea.com/support/downloads>)

2.2.2 Vytvorenie virtuálneho prostredia (odporúčané)

Je odporúčané zadať tento kód do command window.

```
python -m venv venv
source venv/bin/activate
```

UPOZORNENIE: Pred spustením systému je potrebné overiť, či sú ovládače kamery správne nainštalované. Ak používate kameru s Python API, ktoré nie je dostupné cez správcu

balíkov Pythonu, je nutné vytvoriť symbolický odkaz na ovládače kamery alebo skopírovať API priamo do virtuálneho prostredia.

2.2.3 Inštalácia knižníc

Pre spustenie programu je potrebné stiahnuť nasledovné knižnice

```
pip install opencv-python
sudo apt install python3-colcon-common-extensions
pip install matplotlib
pip install pillow
pip install ximea
pip install numpy
pip install torch torchvision --index-url
https://download.pytorch.org/whl/cu118
```

Posledný príkaz je nutné upraviť podľa verzie grafickej karty pre CUDA, verzia 11.8. je kompatibilná s verziou python 3.10 a s grafickou kartou Nvidia 2060. (Odporúča sa pre tento krok využiť umelú inteligenciu prompt: „Mám grafickú kartu XXXX a python verzie 3.10, ako mám upraviť tento príkaz „pip install torch torchvision --index-url https://download.pytorch.org/whl/cu118“, aby som vedel spustiť tento kód.“ (Odporúča sa vložiť kód)).

2.2.4 Inštalácia package (build)

Z adresára, kde sa nachádza Váš package facial_recognition (napríklad ak sa nachádza na ploche)

```
mkdir -p ~/ros2_ws/src
mv ~/Desktop/facial_recognition ~/ros2_ws/src/
cd ~/ros2_ws/
colcon build --packages-select facial_recognition
source install/setup.bash
```

2.2.5 Spustenie Node

```
ros2 run facial_recognition recognition_publisher.py
```

2.2.6 Overenie publikácie

V novom termináli:

```
source ~/ros2_ws/install/setup.bash  
ros2 topic echo /recognition_status
```

2.3 Návod na spustenie – Pracovisko COCOHRIP

Tento návod vás prevedie procesom spustenia aplikácie na rozpoznávanie operátora na pracovisku COCOHRIP.

2.3.1 Spustenie programu

Nakoľko program je na pracovisku plne implementovaný, stačí spustiť command window a zadať nasledujúce príkazy:

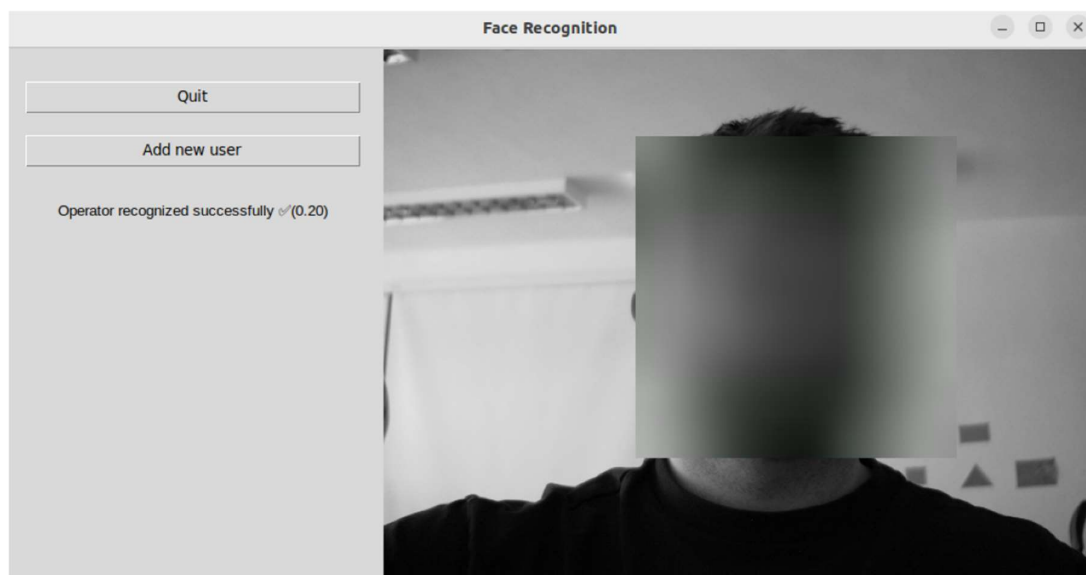
```
Source install/setup.bash  
./src/facial_recognition/run_facial_expression.sh
```

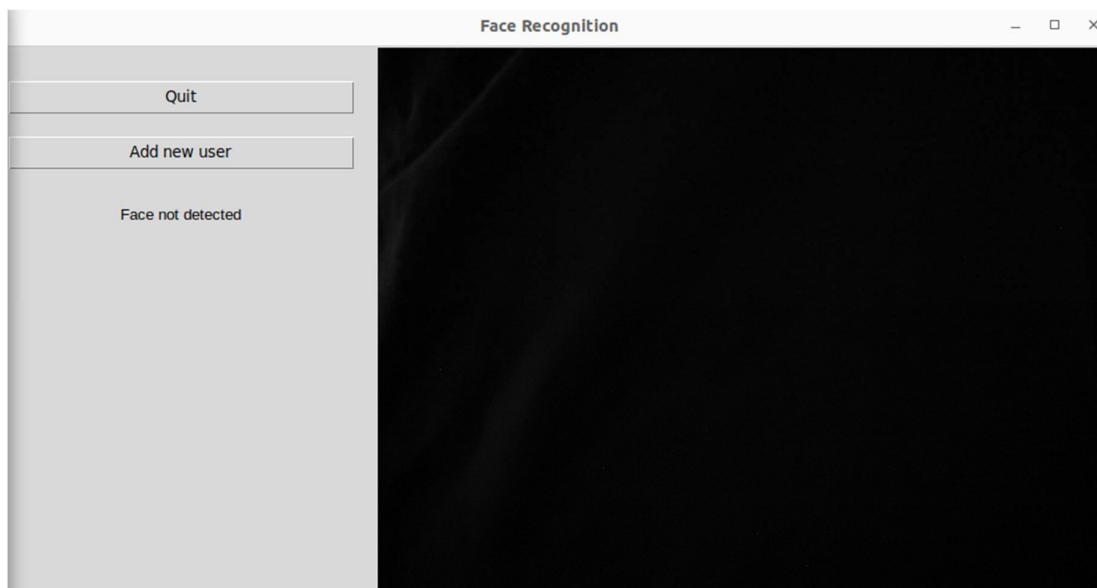
Po zadaní príkazov sa spustí aplikácia a dáta sú publikované na *face_recognition_publisher*. Rovnako ako na Linux aplikácii na publisher je v rovnakom priečinku subscriber.

3 Používateľská príručka

Hlavné okno aplikácie je rozdelené na viacero logických častí:

- Živý náhľad z kamery: V pravej časti rozhrania sa zobrazuje aktuálny obraz z priemyselnej kamery Xiemea, ktorá sníma tvár operátora v reálnom čase.
- Výsledok rozpoznávania: Pod tlačidlami sa nachádza textový prvok, ktorý zobrazuje stav autentifikácie. Zobrazovaný text môže byť napríklad: „Recognizing...“, „Operator recognized successfully“ alebo „Face not detected“.
- Ovládacie prvky: Rozhranie obsahuje viacero tlačidiel, ktoré slúžia na ovládanie systému. Medzi najdôležitejšie patria:
 - „Add new user“ – slúži na zachytenie snímky a pridanie nového používateľa do databázy,
 - „Quit“ – ukončí beh aplikácie.





3.1 Dynamické pridávanie používateľa

Jednou z hlavných funkcií navrhnutého systému je dynamické pridávanie nového operátora bez potreby pretrénovania modelu. Tento proces umožňuje jednoduché rozšírenie databázy o nových ľudí, ktorých identita ešte nie je systému známa.

Po kliknutí na tlačidlo „Add new user“ sa štandardné rozhranie systému zmení a používateľovi sa zobrazí špeciálna sekcia určená pre registráciu novej osoby do systému. V tejto fáze sa ľavý panel rozhrania kompletne zmení a nahradí sa novými ovládacími prvkami a vstupmi.

- Zobrazené prvky v režime pridávania používateľa:
- Textové pole „Enter name:“ – slúži na zadanie mena alebo identifikátora novej osoby.
- Textové pole „Enter password:“ – chráni pridanie používateľa pred zneužitím. Zadáva sa hlavné heslo administrátora (v kóde admin123).
- Tlačidlo „Start taking pictures“ – po jeho kliknutí začne systém snímať tvár nového používateľa.
- Tlačidlo „Back“ – umožňuje návrat späť do hlavného rozhrania bez vykonania zmien.

- Textová notifikácia (status): – priebežne informuje o tom, čo sa práve deje, napr. „Taking pictures in 3...“, „Captured 5/15“, „User added. Reloading recognition model...“
- Progres bar: – vizuálne zobrazuje počet zachytených obrázkov z celkového počtu 15 potrebných snímok.

Dynamické správanie rozhrania:

- Po kliknutí na „Start taking pictures“ sa spustí odpočítavanie (3 sekundy), počas ktorého má používateľ čas sa nastaviť pred kameru.
- Po odpočítaní systém začne automaticky snímať tvár a ukladať jednotlivé obrázky do priečinka podľa zadaného mena.
- Po nasnímaní všetkých 15 snímok sa zobrazí hlásenie o úspechu a začne sa automatické načítanie embeddingov, aby sa nový používateľ stal okamžite rozpoznateľným.
- Po krátkom čase sa systém automaticky vráti späť do hlavného režimu rozpoznávania.

Bezpečnostné kontroly a chyby:

- Ak meno nie je zadané:
„The name input field is empty“
- Ak nie je zadané heslo:
„The password input field is empty“
- Ak je heslo nesprávne:
„Incorrect password!“
- V prípade problémov pri ukladaní obrázkov alebo pri načítaní embeddingov, systém zahlásí chybu a zobrazí upozornenie.


Tento proces robí systém veľmi jednoducho rozšíriteľným – nových operátorov je možné pridávať bez akéhokoľvek zásahu do modelu, čo výrazne šetrí čas. Celý proces je zvládnuteľný priamo cez používateľské rozhranie.

Face Recognition

Enter name:

Enter password:

The name input field is empty




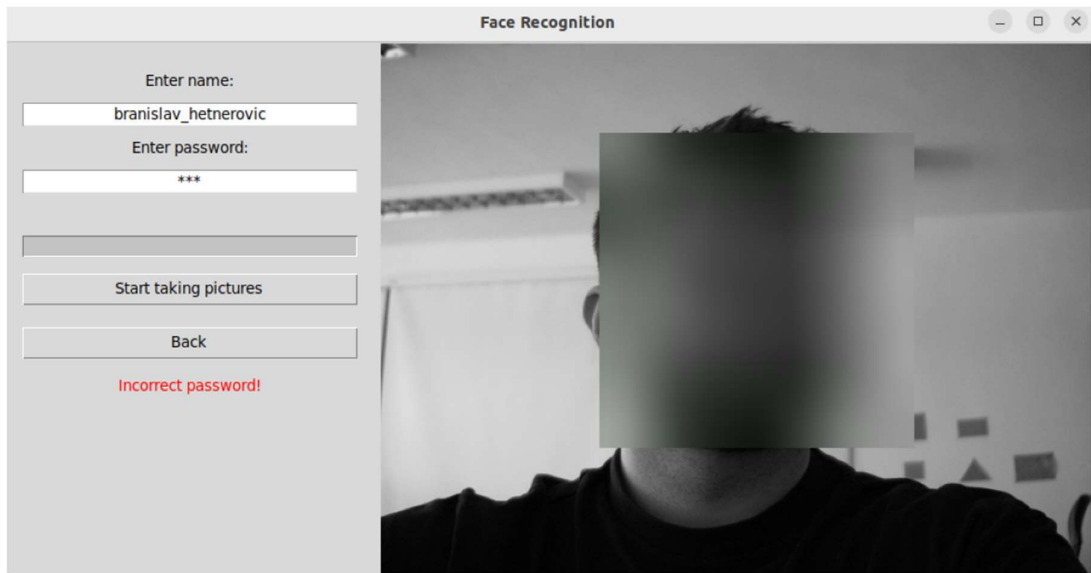
Face Recognition

Enter name:

Enter password:

The password input field is empty

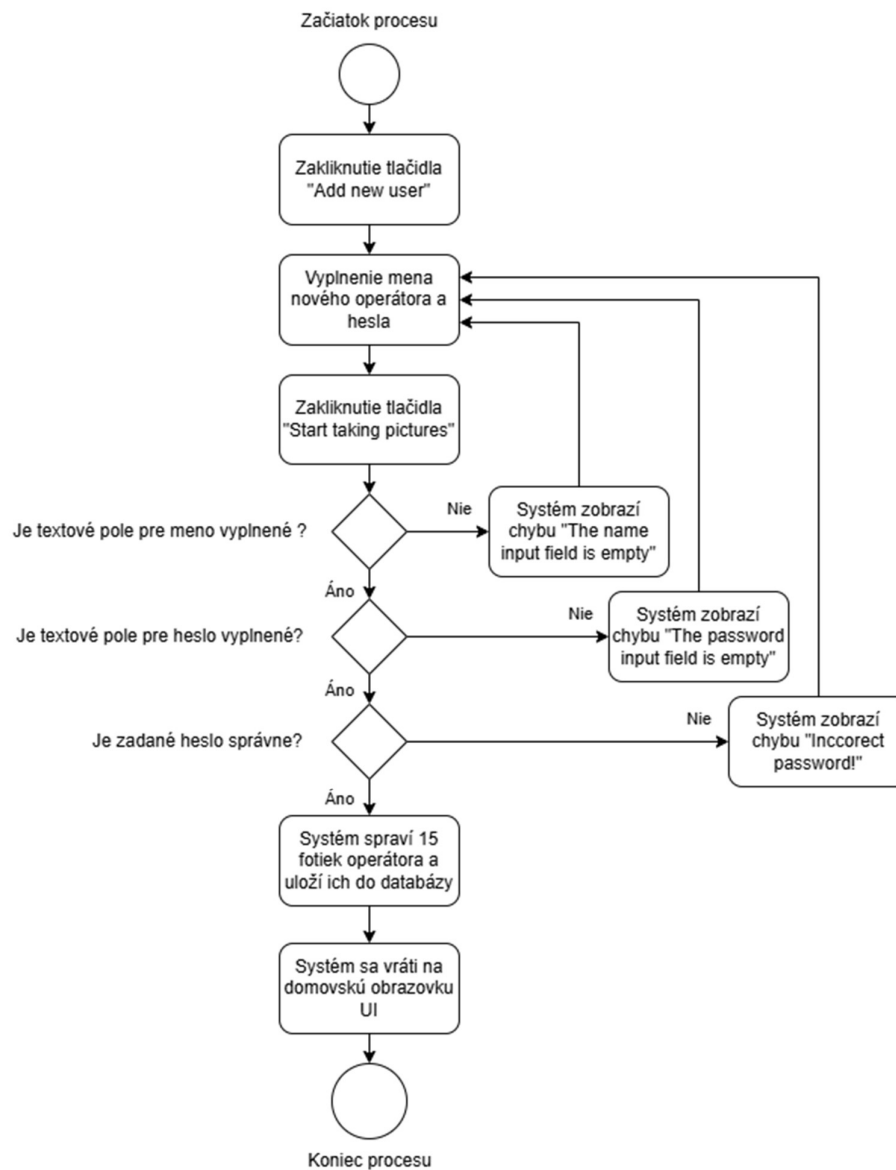




4 Vývojové diagramy

4.1 Postupnosť pridania operátora

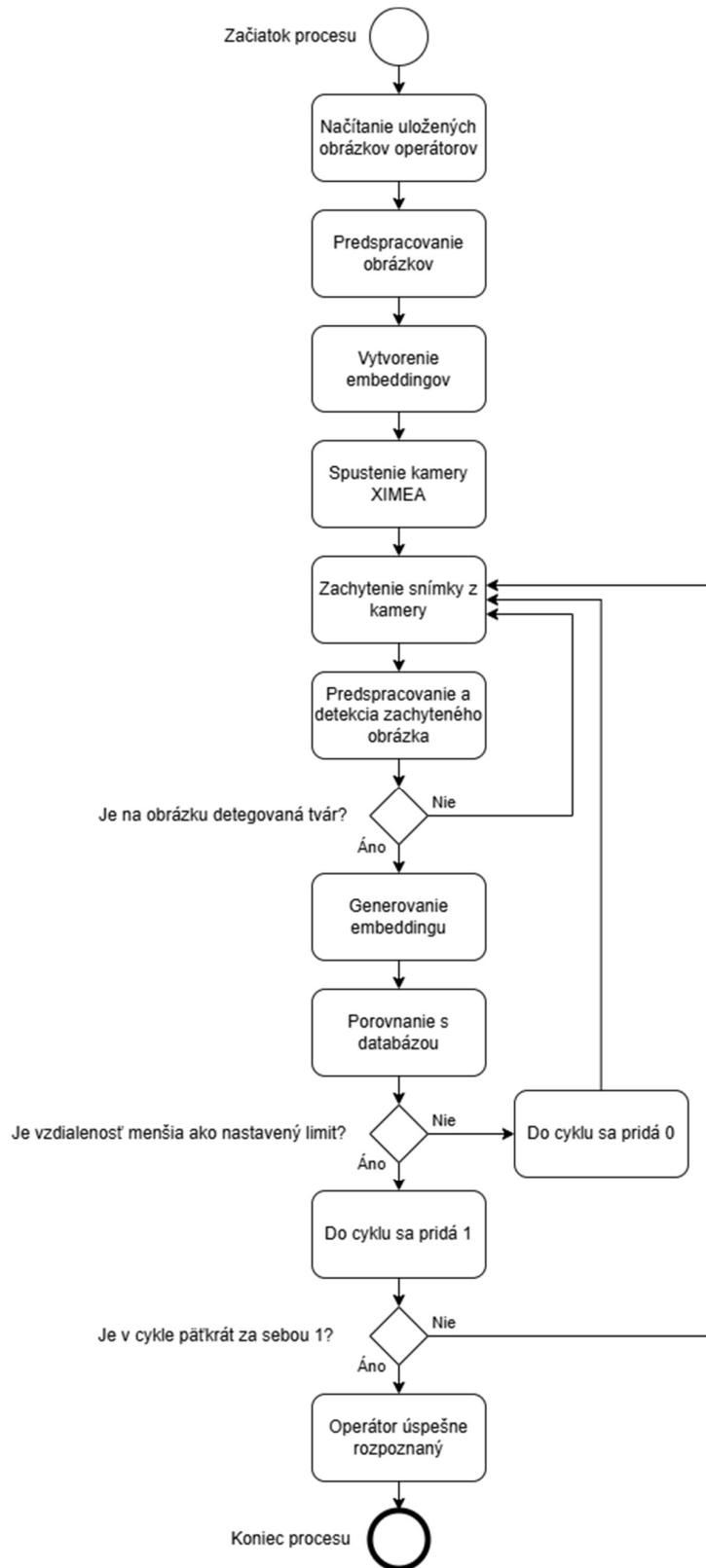
Na zabezpečenie pridania nového operátora do systému bola navrhnutá postupnosť krokov – teda pipeline, ktorá definuje jednotlivé fázy od získania vstupných údajov (meno operátora a heslo) až po ich uloženie do databázy operátorov. Táto sekvencia činností je znázornená na obrázku č. 15.



ID	Názov komponentu	Popis komponentu
1	Začiatok procesu	Používateľ spustí proces pridávania nového operátora v systéme.
2	Zakliknutie tlačidla "Add new user"	Používateľ klikne na tlačidlo pre pridanie nového používateľa.
3	Vyplnenie mena nového operátora a hesla	Do príslušných polí používateľ zadá meno a heslo operátora.
4	Zakliknutie tlačidla "Start taking pictures"	Používateľ spustí proces zachytávania snímok kliknutím na príslušné tlačidlo.
5	Je textové pole pre meno vyplnené?	Systém overí, či bolo vyplnené textové pole pre meno.
6	Systém zobrazí chybu "The name input field is empty"	Ak pole nie je vyplnené, systém zobrazí chybové hlásenie.
7	Je pole pre heslo vyplnené?	Systém overí, či bolo vyplnené pole pre heslo.
8	Systém zobrazí chybu "The password input field is empty"	Ak pole pre heslo nie je vyplnené, zobrazí sa chybové hlásenie.
9	Je heslo správne?	Systém overí správnosť zadaného hesla.
10	Systém zobrazí chybu "Incorrect password!"	Ak je heslo nesprávne, zobrazí sa chybové hlásenie.
11	Systém spraví 15 fotiek operátora a uloží ich do databázy	Systém spraví 15 fotiek operátora po odpočte a uloží ich do databázy do priečinka so zadaným menom operátora.
12	Systém sa vráti na domovskú obrazovku UI	Po úspešnom uložení sa systém vráti na úvodnú obrazovku používateľského rozhrania.
13	Koniec procesu	Proces registrácie nového používateľa je ukončený.

4.2 Postupnosť spracovania obrazu (pipeline systému)

Na zabezpečenie efektívnej autentifikácie operátora na základe tváre bol navrhnutý pipeline – teda sekvenčný proces spracovania vstupných údajov od načítania databázy až po rozhodnutie o identite operátora. Táto sekvencia krokov je zobrazená na obrázku č. 14, kde jednotlivé komponenty zabezpečujú pedspracovanie údajov, ich analýzu pomocou neurónovej siete, a následné porovnanie s databázou.



ID	Názov komponentu	Popis komponentu
1	Začiatok procesu	Používateľ spustí system pre rozpoznávanie tváre operátora.
2	Načítanie uložených obrázkov operátorov	Systém načíta tvárové snímky jednotlivých operátorov, ktoré boli predtým zachytené a uložené do databázy.
3	Predspracovanie obrázkov	Obrázky sú orezané, zmenšené, normalizované a prevedené do vhodného formátu pomocou OpenCV <i>cv2.dnn</i> na zabezpečenie konzistentného vstupu do modelu
4	Vytvorenie embeddingov	Zo všetkých predspracovaných obrázkov sa vygenerujú embeddingy – vektory reprezentujúce črty tváre – pomocou siamskej neurónovej siete.
5	Spustenie kamery XIMEA	Aktivuje sa kamera XIMEA, ktorá sníma aktuálny obraz operátora v reálnom čase.
6	Zachytenie snímky z kamery	Systém zachytí aktuálnu snímku tváre používateľa, ktorá bude porovnávaná s databázou.
7	Predspracovanie a detekcia zachyteného obrázka	Obrázok je predspracovaný a na obrázku sa deteguje tvár.
8	Je na obrázku detegovaná tvár?	Kontroluje sa, či sa na zachytenom obrázku nachádza tvár. Ak nie, pokračuje sa ďalším snímkom.
9	Generovanie embeddingu	Zo zachytenej tváre sa vygeneruje embedding – teda číselná reprezentácia vizuálnych črt.
10	Porovnanie s databázou	Vytvorený embedding sa porovnáva s databázou známych embeddingov pomocou euklidovskej vzdialenosti.
11	Je vzdialenosť menšia ako nastavený limit ?	Ak je vzdialenosť medzi embeddingmi menšia ako nastavený prah, ide o zhodu.
12	Do cyklu sa pridá 0	Systém generuje 0 v prípade, že rozpoznanie bolo neúspešné.
13	Do cyklu sa pridá 1	Systém generuje 1 v prípade, že rozpoznanie bolo úspešné.
14	Je v cycle päťkrát za sebou 1 ?	Systém kontroluje, či bol operátor identifikovaný úspešne 5-krát po sebe. Tým sa znižuje riziko falošnej zhody.
15	Operátor úspešne rozpoznaný	Systém potvrdí identitu operátora ako autentifikovaného a odošle výsledok.
16	Koniec procesu	Proces končí po úspešnom rozpoznaní operátora.

5 Trénovanie vlastného modelu

Tento návod Vás prevedie procesom spustenia aplikácie na trénovanie vlastného modelu nerónovej siete. Je nutné stiahnuť priečinok `trenovanie_neurónovej_siete`, ktorý je možné nájsť v kapitole zdrojové kódy. Tento priečinok obsahuje všetky potrebné súbory pre spustenie programu.

5.1.1 Požiadavky

- Windows 10 alebo 11
- GPU s podporou CUDA (je potrebné overiť verziu kompatibilnú s Vašou grafickou kartou)
- Python 3.8+ (Odporúča sa verzia 3.10)
- Nainštalovaný `georgia` `tech` `face` `dataset` (<https://academictorrents.com/details/0848b2c9b40e49041eff85ac4a2da71ae13a3e4f>)
- Georgia tech face dataset vložený do rovnakého priečinka ako jupyter notebook súbor `trenovanie_SNN.ipynb` (je potrebné dataset manuálne rozdeliť do 3 priečinkov pre test, train a valid)

5.1.2 Inštalácia knižníc

Pre spustenie programu je potrebné stiahnuť nasledovné knižnice

```
pip install opencv-python
pip install matplotlib
pip install pillow
pip install xiapi
pip install numpy
pip install torch torchvision --index-url
https://download.pytorch.org/whl/cu118
```

Posledný príkaz je nutné upraviť podľa verzie grafickej karty pre CUDA, verzia 11.8. je kompatibilná s verziou python 3.10 a s grafickou kartou Nvidia 2060. (Odporúča sa pre tento krok využiť umelú inteligenciu prompt: „Mám grafickú kartu XXXX a python verzie 3.10, ako mám upraviť tento príkaz „`pip install torch torchvision --index-url https://download.pytorch.org/whl/cu118`“, aby som vedel spustiť tento kód. “ (Odporúča sa vložiť kód)).

5.1.3 Spustenie aplikácie

Po nainštalovaní potrebných knižníc a databázy georgia tech face dataset stačí otvoriť súbor `trenovanie_SNN.ipynb` a postupne spúšťať jednotlivé bloky. Váhy pre model budú uložené v priečinku `checkpoints` po dokončení tréovania.