

# Metoda převodu syntetického otisku na fotku

## Biometrické systémy

**Autoři:** Jakub Sadílek (xsadil07)

Lukáš Havlíček (xhavli46)

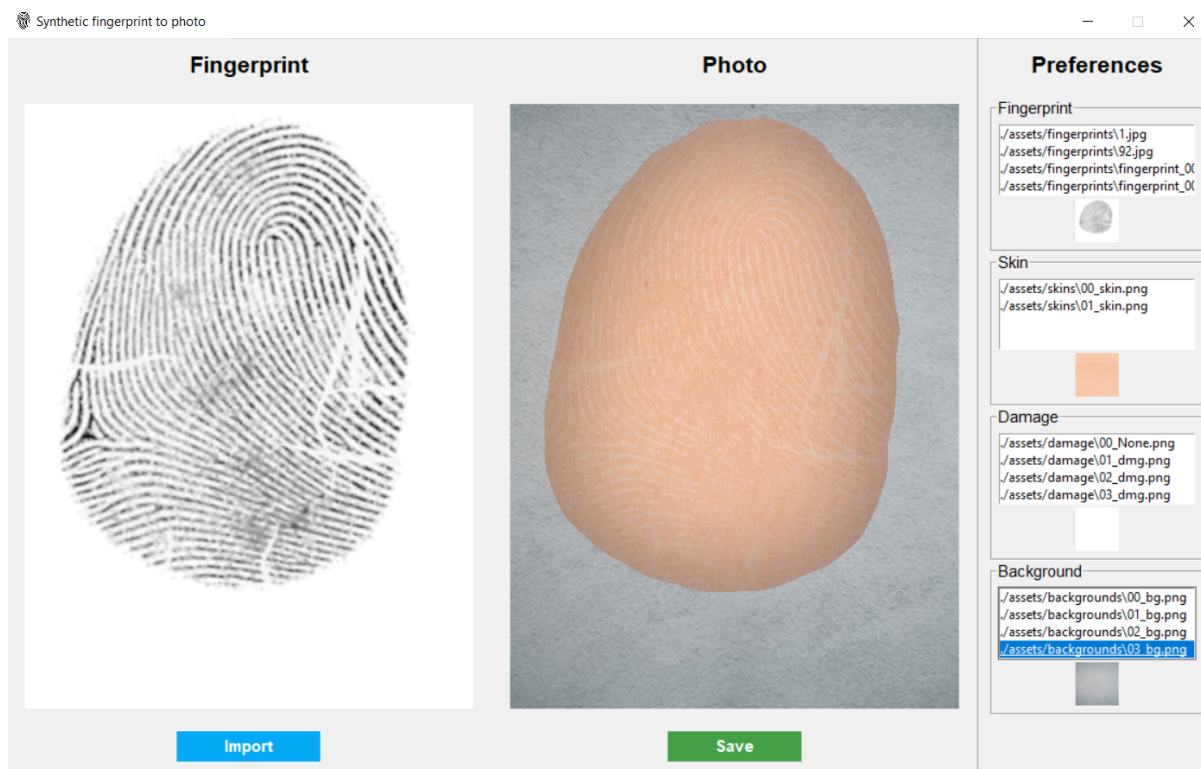
**Datum:** 10.12.2021

### Úvod

Cílem tohoto projektu je vytvořit metodu, která převede obrázek syntetického otisku prstu do podoby jakou by mohly mít kdyby byly vyfoceny. Základní úpravou je změna barev z šedotónového obrázku do odstínů barev kůže. Dále pak byly upraveny okrajové části otisku tak, aby na výsledné "fotografii" působily papilární linie realisticky. Další částí je vygenerování realisticky vypadajícího pozadí. Posledním bodem je pak integrace některých poškození. Jinými slovy na vstupu by měla být fotka jako z optického snímače a na výstupu jako z digitálního mikroskopu.

### Uživatelské rozhraní

Pro snadnější použití programu bylo jako rozšíření implementováno grafické uživatelské rozhraní (viz obrázek 1), ve kterém může uživatel snadno pozorovat rozdíl mezi vstupním syntetickým otiskem a jeho výslednou podobou po transformaci. Na pravé straně okna se pak nachází nabídka, kde uživatel nastavuje parametry programu jako je předpřipravený vstupní syntetický otisk, barva kůže, poškození či pozadí obrázku. Po změně volby se automaticky překreslí výstupní snímek. Uživatel může taktéž nahrát vstupní otisk přes tlačítko "Import" nebo stáhnout výsledek přes pomocí tlačítka "Save". Podporované formáty vstupních obrázků jsou .png, .jpg či .bmp, výsledek je pak ve formátu .png.



Obrázek 1: Grafické uživatelské rozhraní

## Implementace

Jako implementační jazyk byl zvolen Python. Řešení bylo rozděleno na 2 části, generování “fotky” z otisku prstu a na uživatelské rozhraní. Zdrojové kódy pro generování otisku lze nalézt v souboru `main.py` a GUI v souborech `gui.py` a `imageListBox.py`.

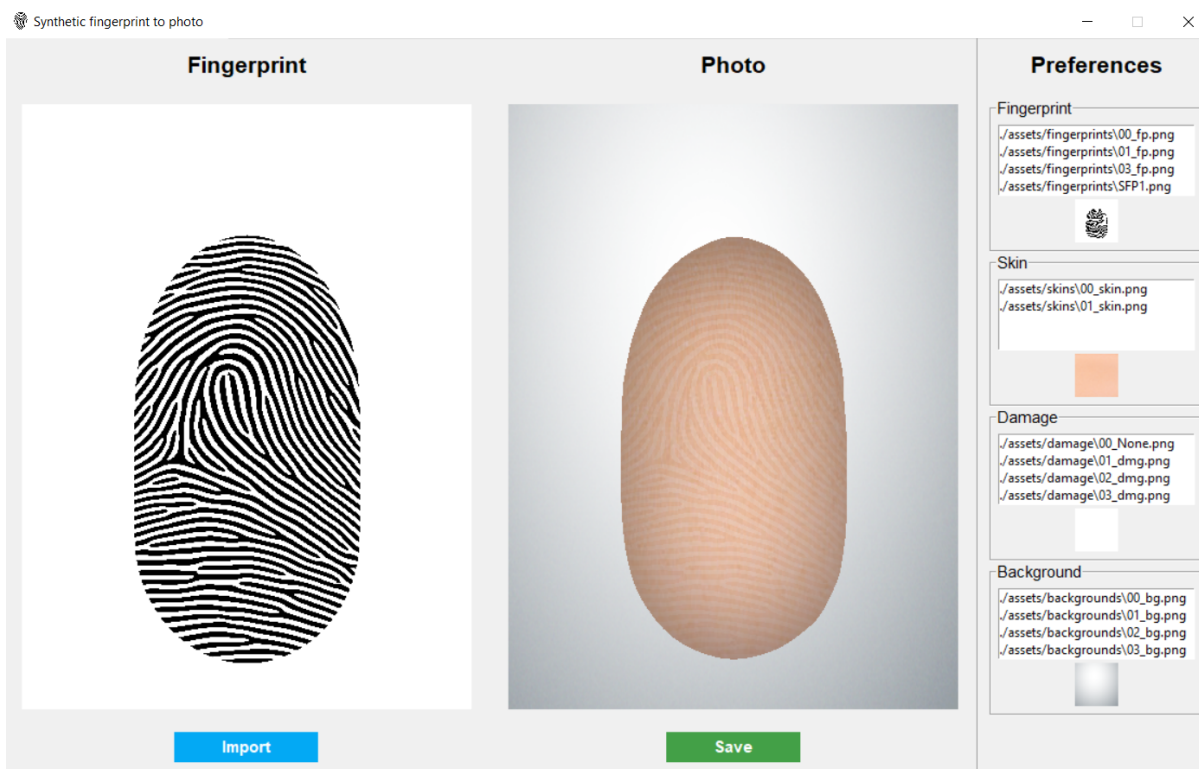
Generování otisku je zajištěno pomocí 4 vstupních souborů: otisk prstu, textura kůže, textura pozadí a maska poškození. Ze vstupního otisku jsou vyrobeny masky pro papilární linie a pro tvar otisku, papilární linie jsou následně namapovány a “vyříznuty” do textury kůže a tvar prstu je namapován na upravenou texturu kůže (lehce ztmavenou), aby linie vypadaly realisticky. Následně je za pomoci masky poškození a znovu jinak upravené textury kůže přidáno poškození na otisk a poté je ztmaven okraj otisku, aby působil 3D. Výsledný otisk je poté vložen na pozadí.

Vstupní soubory jsou očekávány ve složce `assets` a následně ve svých podložkách pojmenovaných podle kategorie, kde je možné přidat vlastní soubory, které aplikace poté načte při spuštění. Podporované formáty těchto dat jsou `.jpg`, `.png` a `.bmp`.

Do složky `background` je možné přidat téměř jakýkoliv obrázek pozadí, který bude umístěn za otisk prstu. Do složky `damage` je možné si přidat vlastní masku poškození, která musí být nakreslena černou barvou (ideálně s maximální ostroť) a pozadí musí být vyplněno bílou barvou. Do složky `fingerprints` je možné vložit vlastní obrázky syntetických otisků prstů, kde se opět očekává otisk černou barvou na bílém pozadí (pro masku je brána hodnota velice světle šedé, aby byly správně namapovány i “poškozené” otisky, kde pozadí není čistě bílé). Do složky `skins` je možné vložit vlastní texturu kůže, na kterou se následně mapují papilární linie a tvar prstu.

## Instalace a spuštění

Jako první krok ke spuštění projektu je nutné nainstalovat potřebné knihovny pomocí příkazu `pip install -r requirements.txt` v kořenovém adresáři projektu, který následně provede instalaci podle souboru `requirements.txt` (*Numpy, Pillow, Scipy*). Taktéž je potřeba knihovna Tkinter, která by měla být součástí Pythonu. Nakonec je možné spustit aplikaci pomocí příkazu `python gui.py` (viz obrázek 2).



Obrázek 2: Úvodní rozhraní aplikace

## Závěr

Výslednou realizací je python aplikace s grafickým uživatelským rozhraním, ve které může uživatel snadno volit vstupní parametry a okamžitě vidí změny jak co se týče vstupního otisku, tak co se týče výsledného obrázku. Díky využití reálné textury kůže vypadá výsledný otisk téměř realisticky.