

DACTYLOS

# Technická dokumentácia

Verzia 1.0

Bc. Róbert Mokráš  
25.1.2015

## Obsah

1 Program .....	3
1.1 Požiadavky .....	3
1.2 Spustenie programu .....	3
2 Funkcie.....	3
2.1 Hlavné Funkcie .....	3
2.2 Get a Set funkcie.....	4
2.3 Pomocné funkcie .....	4

# 1 Program

## 1.1 Požiadavky

Minimálne požiadavky pre správny beh programu:

- Nainštalovaný programovací jazyk C++
- Nainštalovaná knižnica OpenCV
- Nainštalovaný Operačný systém
- Dostatok voľnej pamäte RAM

## 1.2 Spustenie programu

Program sa skladá z jedného hlavičkového súboru `preprocessing.hpp` a jedného zdrojového súboru `preprocessing.cpp`. Tieto súbory slúžia iba na vloženie do iných zdrojových súborov, nie sú spustiteľné samostatne.

# 2 Funkcie

## 2.1 Hlavné Funkcie

Všetky hlavné funkcie sú ako `public`.

- Program má 2 typy konštruktorov

`Preprocessing(cv::Mat & image);`

`Preprocessing(cv::Mat & image, int horizontal_resolution, int vertical_resolution);`

Prvý konštruktor má iba jeden argument a to vstupný obrázok, ktorý si upraví do čiernobielej podoby a dá implicitné rozlíšenie 500 Dpi. Druhý konštruktor pracuje na tom istom princípe, len má ešte dva dodatočné argumenty a to vertikálne a horizontálne rozlíšenie obrázka.

- `Segmentation(cv::Mat & image);`

Funkcia oddeľuje odtlačok prsta od pozadia. Vzniká v nej maska odtlačku prsta, ktorá sa uloží do premennej `mask`.

- `OrientationMap(cv::Mat & image);`

Funkcia vypočítava lokálne smery papilárnych línií a priemeruje ich pomocou Gaussového filtra. Vytvorí sa smerová mapa odtlačku prsta.

- `GaborFilter(cv::Mat & image);`

Funkcia upravuje odtlačok prsta. Vstupom do tejto funkcie je smerová mapa.

- `Binarization(cv::Mat & image);`

Funkcia prideli jednotlivým pixelom hodnotu 0 alebo 255.

- `ThinningImage(cv::Mat & image);`

Funkcia zúži papilárne línie na hrúbku jeden pixel.

## 2.2 Get a Set funkcie

Všetky get a set funkcie sú ako inlinepublic funkcie a pristupujú k súkromným premenným v triede.

- `inlineintGet_sizeofBlockGabor()` – vráti veľkosť bloku pri Gáborovom filtri
- `inlineintGet_sizeofBlockOrientation()` – vráti veľkosť bloku pri orientácii papilárnych línií
- `inlinedoubleGet_sigma()` – vráti sigmu z Gáborovho filtra
- `inlinedoubleGet_lambda()` – vráti lambdu z Gáborovho filtra
- `inlinedoubleGet_gamma()` – vráti gammu z Gáborovho filtra
- `inlinevoidSet_sizeofBlockGabor(intsize)` – nastaví veľkosť bloku pri Gáborovom filtri
- `inlinevoidSet_sizeofBlockOrientation(intsize)` – nastaví veľkosť bloku pri orientácii papilárnych línií
- `inlinevoidSet_sigma(double sigma)` – nastaví sigmu z Gáborovho filtra
- `inlinevoidSet_lambda(doublelambda)` – nastaví lambdu z Gáborovho filtra
- `inlinevoidSet_gamma(gamma)` – nastaví gammu z Gáborovho filtra

## 2.3Pomocné funkcie

Všetky pomocné funkcie sú private

- `intsupplement_rows(int W, intW_row);`  
funkcia dopočíta zvyšné riadky v odtlačku prsta
- `intsupplement_cols(int W, intW_col);`  
funkcia dopočíta zvyšné stĺpce v odtlačku prsta
- `voidwiener_to_mask(intblock_size_x, intblock_size_y);`  
funkcia aplikuje na masku odtlačku wienerov filter
- `voidGuoHall(cv::Mat & image, intiteration);`  
funkcia, ktorá je potrebná pre výpočty zúženia papilárnych línií vo funkcii `ThinningImage()`