

Przetwarzanie obrazów cyfrowych i sekwencji wideo w OpenCV

Ćwiczenia 1

Marcin Ciecholewski

Klasyfikacja ćwiczeń

Zadania: 60% oceny końcowej, Projekt: 40% oceny końcowej.

Zwolnienie z egzaminu umożliwia uzyskanie 71% z Zadań.

Oceny będą wystawiane na podstawie następującej punktacji:

0-50 2.0

51-60 3.0

61-70 3.5

71-80 4.0

81-90 4.5

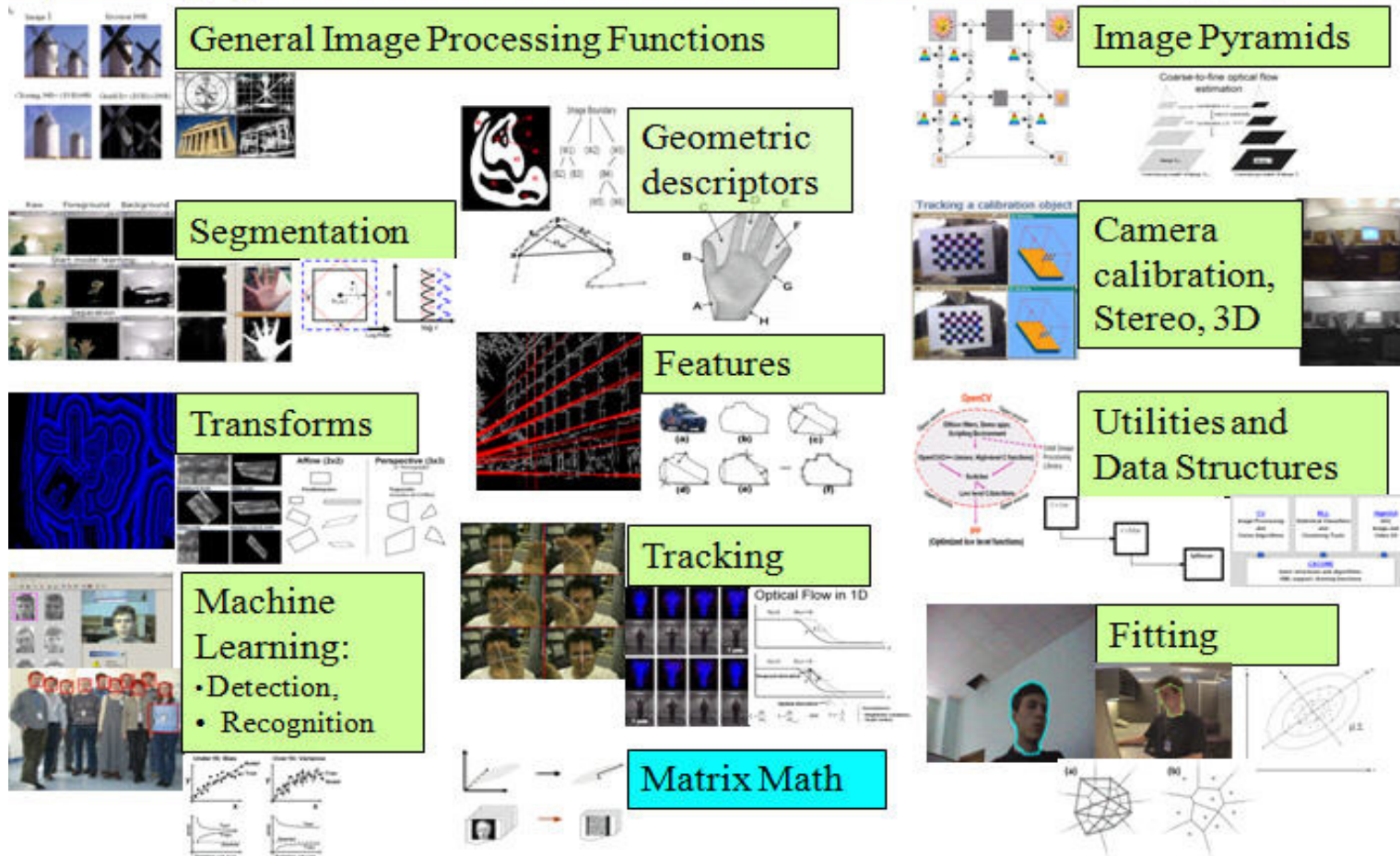
91-100 5.0

Biblioteka OpenCV

OpenCV Overview: > 500 functions

opencv.willowgarage.com

Robot support



Program w OpenCV i C++: wprowadzenie

- Należy utworzyć nowy projekt w środowisku Visual C++: **New Project**→**Visual C++**
- Następnie wybrać: **Empty Project**
- W menu, mamy do dyspozycji następujące konfiguracje: **Debug|x86**, **Release|x86**, **Debug|x64**, **Release|x64**. Zaznaczmy: **Debug|x64**
- Następnie z menu kontekstowego wybieramy: **Properties**

Program w OpenCV i C++: wprowadzenie

- Wybieramy: **VC++ Directories**, po czym należy wpisać dodatkowe katalogi z plikami nagłówkowymi, czyli **Include Directories:** C:\opencv\build\include
- Wybieramy **Linker (Konsolidator)**, dalej zakładkę **General (Ogólne):**

C:\opencv \build\x64\vc15\lib

Po czym należy podać jedną dodatkową bibliotekę w **Additional Dependencies (Dodatkowe zależności)**, czyli: opencv_world340d.lib

Program w OpenCV i C++: wprowadzenie

- ❑ Wracamy do kodu projektu. Wybieramy: **Solution Explorer** a następnie za pomocą menu kontekstowego dodamy nowy plik z rozszerzeniem .cpp do projektu (np. main.cpp)
- ❑ Kod źródłowy:

```
#include<opencv2/highgui/highgui.hpp>
:
:
#include<iostream>
#include<conio.h>    // may have to modify this line if not using Windows

using namespace cv;
using namespace std;
```

Program w OpenCV i C++: wprowadzenie

Uwagi:

Wszystkie klasy i funkcje biblioteki OpenCV znajdują się w przestrzeni nazw 'cv' dlatego został wykorzystany alias

`using namespace cv`

Inne podejście to zastosowanie operatora zakresu, czyli (::)

□ Kod źródłowy:

```
#include<opencv2/highgui/highgui.hpp>
...
#include<iostream>
#include<conio.h>    // may have to modify this line if not using Windows

using namespace cv;
using namespace std;

int main() {
```

Program w OpenCV i C++: wprowadzenie

□ Kod źródłowy cd.

```
// Utworzenie macierzy przechowującej obraz (o dowolnych, skończonych wymiarach)
```

```
Mat image;
```

```
// Warto przeczytać w dokumentacji o możliwościach klasy: Mat
```

http://opencv.jp/opencv-2.2_org/cpp/core_basic_structures.html#mat

http://docs.opencv.org/doc/tutorials/core/mat_the_basic_image_container/mat_the_basic_image_container.html

```
// Inicjalizacja przechwytywania
```

```
VideoCapture cap;
```


Program w OpenCV i C++: wprowadzenie

□ Kod źródłowy cd.

```
// Utworzenie macierzy przechowującej obraz (o dowolnych, skończonych wymiarach)
```

```
Mat image;
```

```
// Warto
```

Uwagi:

Stosując operator zakresu, mamy następujący zapis:

```
cv::Mat image;
```

[http://](http://docs.opencv.org/doc/tutorials/core/mat_the_basic_image_container/mat_the_basic_image_container.html)

[2.2](http://docs.opencv.org/doc/tutorials/core/mat_the_basic_image_container/mat_the_basic_image_container.html)

We wcześniejszych wersjach OpenCV stosowano strukturę `IplImage`. W najnowszych wersjach OpenCV zaleca się używanie klasy `cv::Mat`

http://docs.opencv.org/doc/tutorials/core/mat_the_basic_image_container/mat_the_basic_image_container.html

```
// Inicjalizacja przechwytywania
```

```
VideoCapture cap;
```

Program w OpenCV i C++: wprowadzenie

□ Kod źródłowy cd.

```
// Utworzenie macierzy przechowującej obraz (o dowolnych, skończonych wymiarach)
```

```
Mat image;
```

```
// Warto przeczytać w dokumentacji o możliwościach klasy: Mat
```

http://opencv.jp/opencv-2.2_org/cpp/core_basic_structures.html#mat

http://docs.opencv.org/doc/tutorials/core/mat_the_basic_image_container/mat_the_basic_image_container.html

```
// Inicjalizacja przechwytywania
```

```
VideoCapture cap;
```

VideoCapture : klasa umożliwiająca przechwytywanie obrazów z plików video lub kamer

Program w OpenCV i C++: wprowadzenie

- Kod źródłowy cd.

```
cap.open(1);
```

Program w OpenCV i C++: wprowadzenie

- Kod źródłowy cd.

```
cap.open(1);
```

cap.open(0) otwiera wbudowaną kamerę w laptopie. Jeśli oprócz kamery w laptopie jest także podłączona kamera zewnętrzna to **cap.open(1)** otworzy kamerę zewnętrzną a **cap.open(0)** otworzy wbudowaną kamerę laptopa

Program w OpenCV i C++: wprowadzenie

□ Kod źródłowy cd.

```
// Utworzenie okna w celu pokazania obrazu z kamery
namedWindow("window", CV_WINDOW_AUTOSIZE);
|
// Ustalenie rozmiaru okna o wymiarach (640, 480)
double dWidth = cap.set(CV_CAP_PROP_FRAME_WIDTH, 640);
double dHeight = cap.set(CV_CAP_PROP_FRAME_HEIGHT, 480);

while(1){

    // kopiowanie obrazu z kamery

    cap>>image;

    // wyświetlanie skopiowanego obrazu z kamery w oknie

    imshow("window", image);

    // opóźnienie o wartość 15 milisekund (proszę sprawdzić co będzie jeśli nie ma opóźnienia?)
    waitKey(15);

}
return 0; }
```

Program w OpenCV i C++: wprowadzenie

□ Kod źródłowy cd.

```
// Utworzenie okna w celu pokazania obrazu z kamery  
namedWindow("window", CV_WINDOW_AUTOSIZE);
```

Wprowadzić do kodu drobne modyfikacje:

a) Obsługę błędu jeśli inicjalizacja przechwytywania z kamery nie będzie pomyślna

b) Umożliwić wyjście z programu np. naciśnięciem klawisza ,ESC'

```
imshow("window", image);
```

```
// opóźnienie o 15 milisekund (proszę sprawdzić co będzie jeśli nie ma opóźnienia?)  
waitKey(15);
```

```
}
```

```
return 0; }
```