
	Politechnika Bydgoska im. J. J. Śniadeckich Wydział Telekomunikacji, Informatyki i Elektrotechniki <b>Zakład Systemów Teleinformatycznych</b>		
<b>Przedmiot</b>	Przetwarzanie obrazów		
<b>Prowadzący</b>	mgr inż. Grzegorz Czczot		
<b>Temat</b>	Wstęp do przetwarzania obrazów		
<b>Student</b>			
<b>Nr lab.</b>	1	<b>Data wykonania</b>	
<b>Ocena</b>		<b>Data oddania spr.</b>	

## 1. Cel ćwiczenia

Podczas laboratorium zostanie przeprowadzona instalacja środowiska niezbędnego do wykonywania zadań, a także uruchomienie prostych przykładów, które umożliwią poznanie wykorzystywanej biblioteki OpenCV.

## 2. Przebieg ćwiczenia

Na początku należy pobrać: JDK (dla odpowiedniej wersji systemu), pakiet instalacyjny programu NetBeans oraz bibliotekę OpenCV.

Na początku należy zainstalować JDK. Najlepiej nie zmieniać domyślnie ustalonej ścieżki pod którą zostanie przeprowadzona instalacja. Następnie zainstalować NetBeans. Jeśli instalator nie wykryje ścieżki do JDK, należy ją wskazać.

Po zakończeniu instalacji JDK i NetBeans, należy rozpakować bibliotekę OpenCV bezpośrednio na dysku C. Taka lokalizacja pozwoli uniknąć nam wielu problemów w dalszej części konfiguracji. Należy pamiętać, że biblioteki OpenCV nie należy przenosić w inne miejsce. Takie działanie spowoduje, że wszystkie poniższe kroki będzie trzeba wykonać ponownie.

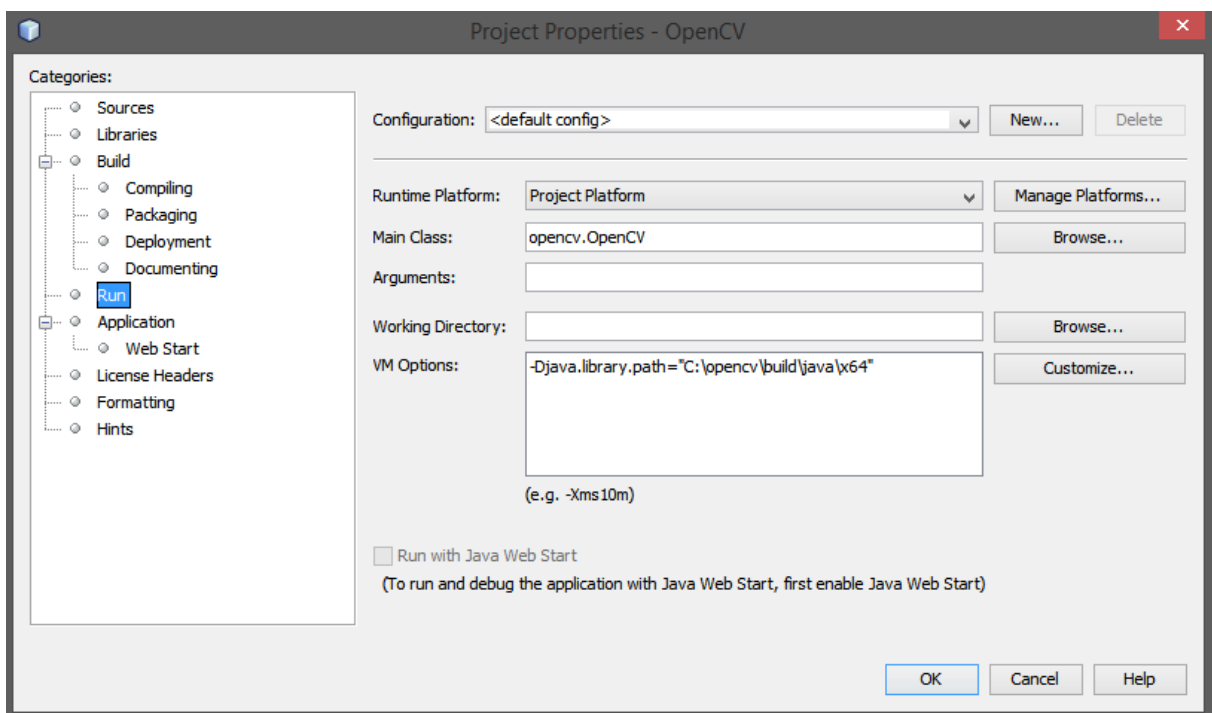
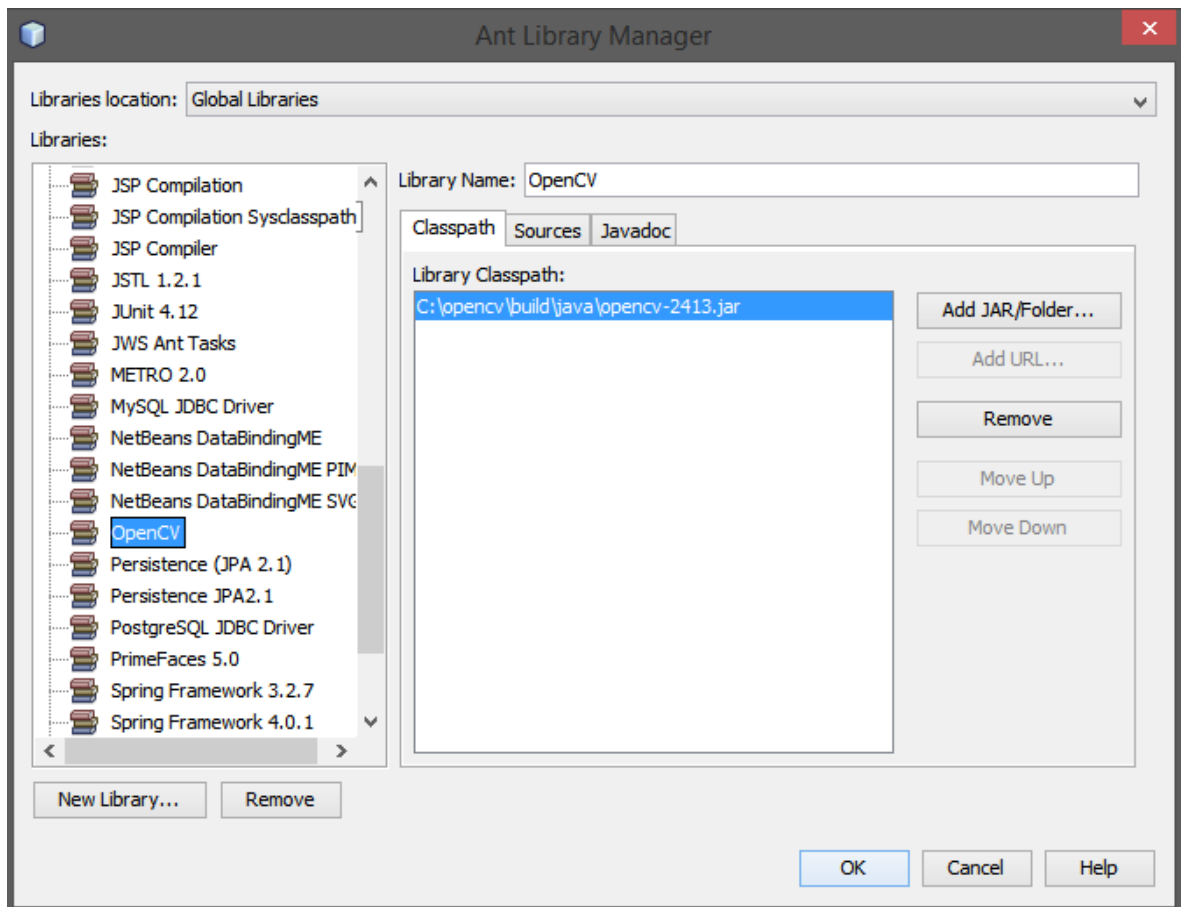
Po poprawnym zainstalowaniu programu NetBeans, należy go uruchomić. Włączyć konfigurację bibliotek (Tools ☉ Libraries). Utworzyć nową bibliotekę przez przycisk „New Library...”. Pojawi się ona na liście po lewej stronie.

Wybrać nową bibliotekę z listy i dodać ścieżkę do „C:\opencv\build\java\opencv-\*\*\*.jar”, gdzie \* oznacza nazwę pliku jar znalezionej pod tym adresem. Zatwierdzić.

Następnie utworzyć nowy projekt (wybierając Java Application). Pojawi się on w oknie „Projects”. Rozwinąć drzewo nowego projektu, nacisnąć „Libraries ☉ Add library...” i wybrać bibliotekę OpenCV z listy. Następnie wybrać właściwości projektu (Properties ☉ Run) i w oknie VM Options wpisać tekst:

-Djava.library.path="C:\opencv\build\java\x64"

gdzie zostanie podana ścieżka do pliku .dll. W zależności od wersji systemu operacyjnego należy wpisać x64 lub x86.



W kodzie programu poza metodą *main*, ale wewnątrz utworzonej klasy, należy umieścić następujący kod, który jest niezbędny do korzystania z biblioteki OpenCV.

```
static {  
    System.loadLibrary(Core.NATIVE_LIBRARY_NAME);  
}
```

Następnie należy dodać tzw. *importy*. Można je dodać szybko przez kliknięcie na żaróweczkę, która pojawia się przy numerze linii. Po ich dodaniu należy uruchomić program naciskając zielony trójkąt w górnym menu lub korzystając z klawisza funkcyjnego F6.

Jeśli w oknie *Output* jest widoczny napis **BUILD SUCCESSFUL**, można przejść do wykonywania zadań.

### 3. Zadania do samodzielnego wykonania

#### 3.1. Zadanie 1.

Oto przykładowy kod, który tworzy macierz i wypisuje ją na ekranie. Należy ten kod wkleić do metody *main* i uruchomić program. Sprawdzić, czy macierz została poprawnie wypisana.

```
Mat img = new Mat(3, 3, CvType.CV_8U);  
int row = 0, col = 0;  
img.put(row, col, 1, 2, 3, 3, 5, 6, 6, 7, 8);  
System.out.println(img.dump());
```

#### 3.2. Zadanie 2.

Zmienić treść programu tak, aby wczytywał obraz (np. *lena.jpg*) do pamięci i zapisywał go w niezmienionej postaci w głównym katalogu projektu.

Aby znaleźć odpowiednie metody, należy posługiwać się dokumentacją biblioteki OpenCV.

#### 3.3. Zadanie 3.

Zmienić treść programu tak, aby wczytywał obraz (np. *lena.jpg*) do pamięci i wyświetlał go w okienku.

Uwaga: należy napisać swoją metodę, która umożliwi wyświetlanie w okienku.

#### 3.4. Zadanie 4.

Zmienić treść programu tak, aby został wyświetlony ten sam obraz, ale w skali szarości.

### 3.5. Zadanie 5.

Spróbować wczytać do programu i wyświetlić pliki o następujących rozszerzeniach:

- .BMP
- .JPG
- .PNG
- .GIF

Czy udało się wyświetlić wszystkie pliki? Czy został zgłoszony jakiś błąd? Udzielić odpowiedzi na pytania w sprawozdaniu.

### 3.6. Zadanie 6.

a) Zmienić kod programu tak, aby na wczytanym obrazie umieszczał dowolny napis i wyświetlał obraz ze zmianami w nowym oknie.

b) Zmienić w kodzie programu właściwości napisu: jego wielkość, położenie, kolor i rodzaj czcionki i wyświetlić obraz ze zmianami w nowym oknie.

### 3.7. Zadanie 7.

a) Zmienić kod programu tak, aby na wczytanym obrazie umieszczał okrąg, prostokąt i linię, a obraz ze zmianami wyświetlał w nowym oknie.

b) Zmienić w kodzie programu właściwości figur geometrycznych, ich wielkość, kolor, położenie oraz grubość linii. Obraz ze zmianami wyświetlić w nowym oknie.

Uwaga: należy korzystać z metod biblioteki OpenCV.

## 4. Sprawozdanie

W sprawozdaniu należy zawrzeć:

- wypełnioną tabelę z początku instrukcji;
- skopiowane istotne części kodu programów napisanych w trakcie zajęć;
- opis wykonanych zadań ze zrzutami ekranu;
- własne spostrzeżenia jako wnioski.