

Raport

Projekt: Detekcja i śledzenie obiektów w czasie rzeczywistym z
kamer wideo

Przedmiot: Programowanie w języku Python II

Mateusz Kamyk

Nr indeksu: 54517

Rok: I, semestr: II, grupa: 3

Data wykonania: 30.06.2024

Spis treści

I. Cel zadania.....	3
II. Wykonane założenia.....	3
III. Znane problemy, braki.....	3
IV. Opis tworzenia projektu.....	3
V. Przykładowe działanie programu.....	4
VI. Testowanie.....	4
VII. Wnioski.....	4

I. Cel zadania

Stworzenie programu w języku Python, który ma na celu wykryć wskazany przedmiot i śledzić go, gdy jest w zasięgu kamery. Ma to się odbywać w czasie rzeczywistym.

II. Wykonane założenia

Program posiada następujące funkcje:

- Oparcie programu na sprawdzonym i bardzo popularnym rozwiązaniu – wykorzystaniu biblioteki OpenCV.
- Możliwe jest określenie obiektu do wykrycia; kod pod tym względem jest uniwersalny
- Możliwe jest określenie kamery, która ma być wykorzystana do powierzonego zadania
- Program działa w trybie interaktywnym
- Po wywołaniu programu otwiera się automatycznie okno z widokiem kamery
- W oknie widoku kamery zaznaczony jest biały kształt obiektu, który jest śledzony
- Wykrywanie i śledzenie obiektu występuje w czasie rzeczywistym
- Stworzenie filmu demo.mp4, które przedstawia działanie programu

III. Znane problemy, braki

Stworzony program do wykrywania i śledzenia bieków nie jest pozbawiony wad, które w przyszłości zostaną naprawione:

- Wyświetlanie białego obrębu wykrywanego obiektu, nawet gdy nie jest on w polu widzenia
- Dość małe okno kamery 'Video'.

IV. Opis tworzenia projektu

Pracę nad kodem wykonywano w Visual Studio Code. Po zakończeniu pracy kod został wprowadzony na GitHuba, gdzie następnie dokonano końcowych poprawek. Podczas pracy korzystano głównie z dokumentacji Pythona oraz użytych bibliotek (tutaj szczególnie opencv-python), a także <https://stackoverflow.com/>.

V. Przykładowe działanie programu

Przykładowe uruchomienie programu, z wskazaniem pliku, w którym program ma umieścić podsumowanie:

```
./src/main.py -filename ../../image/black_mouse.png --camera /dev/video2
```



To close the "Video" window please press ESC.

[demo/demo.mp4](#) przedstawia działanie programu w czasie rzeczywistym.

VI. Testowanie

Testowanie stworzonego programu przebiegało manualnie. Obejmowało najpopularniejsze przypadki użytkowe programu.

VII. Wnioski

Praca nad powyższym projektem była bardzo owocna w poznanie nowych rozwiązań oraz technik tworzenia kodu w Pythonie. Nauka korzystania i integrowania image-processingu z pewnością wpłynie korzystnie na umiejętności programistyczne.