

Przedmiot: Wirtualna rzeczywistość i systemy stereowizyjne

Temat projektu: Skaner okładek książek.

Spis treści:

## Spis treści

<u>WSTĘP.....</u>	<u>4</u>
<u>KONCEPCJA PROPONOWANEGO ROZWIĄZANIA.....</u>	<u>4</u>
<u>ABSTRAKT.....</u>	<u>3</u>
<u>PODSUMOWANIE.....</u>	<u>6</u>
<u>LITERATURA.....</u>	<u>8</u>
<u>DODATEK A: OPIS OPRACOWANYCH NARZĘDZI I METODY POSTĘPOWANIA.....</u>	<u>8</u>
<u>DODATEK B: REALIZACJA PROPONOWANEGO ROZWIĄZANIA.....</u>	<u>9</u>
<u>DODATEK C. OPIS INFORMATYCZNY PROCEDUR.....</u>	<u>9</u>
<u>DODATEK D. SPIS ZAWARTOŚCI DOŁĄCZONYCH NOŚNIKÓW (DYSKIETEK, CD ROMU).....</u>	<u>9</u>

Wykonali: Bartłomiej Bułat, Tomasz Drzewiecki

5 rok *Inf.Stos*

konsultant: *dr inż. Jaromir Przybyło*

## 1. Abstrakt

Praca miała na celu stworzenie aplikacji i biblioteki rozpoznającej książki po okładce. Biblioteka na podstawie zbioru okładek określa czy i jaka książka znajduje się w aktualnej klatce filmu lub obrazu z kamery. Każda okładka ma wyliczone deskryptory i ich punktu kluczowe za pomocą algorytmu SURF[1]. Dopasowanie deskryptorów wykonywane jest w oparciu o algorytm dopasowania FLANN<sup>1</sup>. Na podstawie ilości i jakości dopasowania wybierana jest odpowiednia książka z bazy. Dodatkowo wykonywane jest przekształcenie homograficzne, odnajdujące dokładne położenie książki w ramce obrazu. Biblioteka potrafi również dodać do bazy okładkę książki z ramki filmu po wcześniejszej korekcji perspektywy.

Cel został osiągnięty, powstała aplikacja umożliwiająca wyszukiwanie książek w sekwencjach filmowych lub na materiale z kamery odnaleźć i opisać książkę z zadowalającą dokładnością.

Słowa kluczowe: rozpoznawanie książek, SURF, FLANN, deskryptory obrazu, dopasowanie obrazów, wyszukiwanie obrazem.

## 2. Wstęp

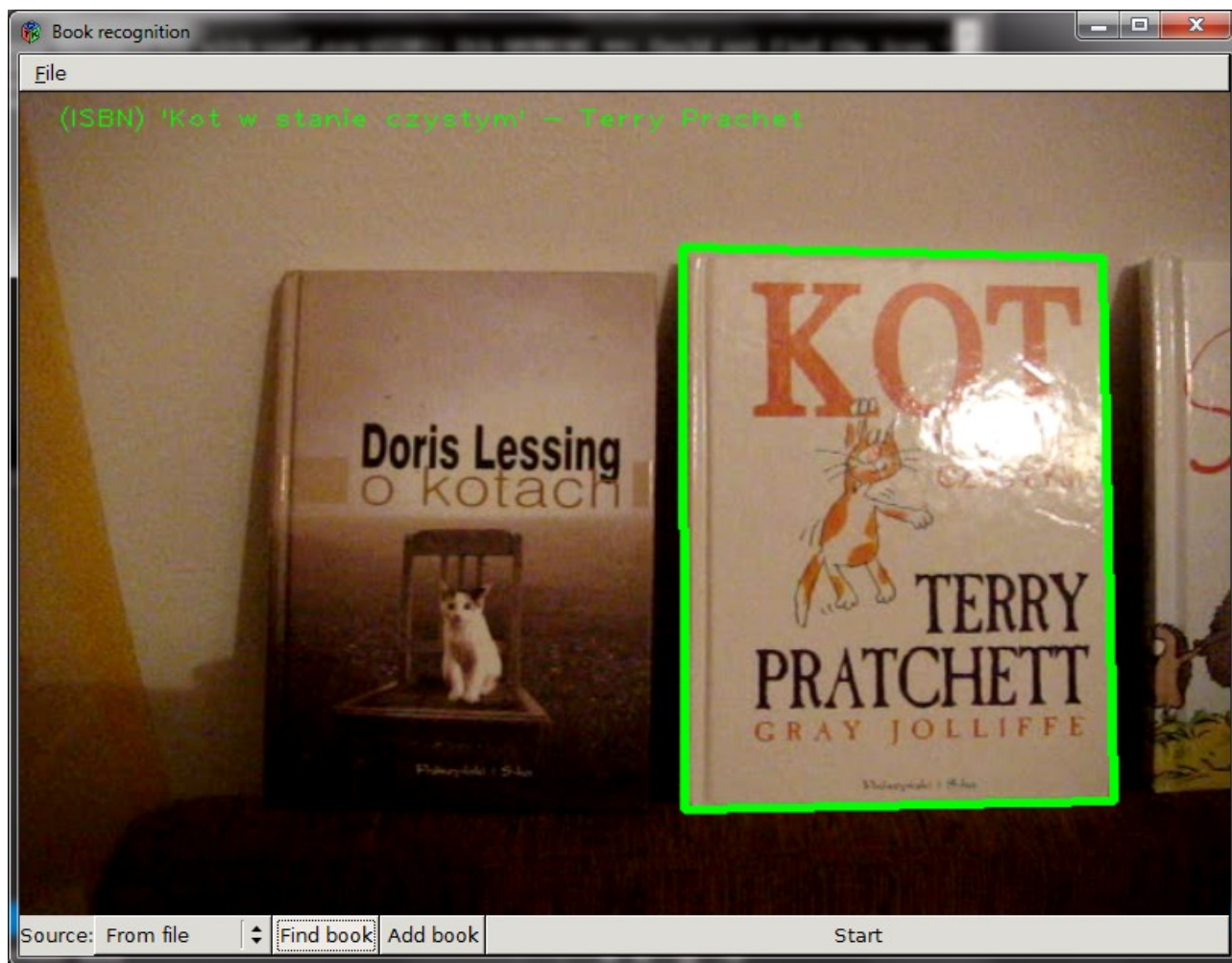
Celem projektu była realizacja aplikacji i biblioteki wyszukującej i rozpoznającej na obrazie książki po okładce. Praktycznym problemem w tego typu projekcie jest określenie cech na podstawie których dany obiekt ma być znajdowany/rozpoznany. Istnieje wiele możliwości opisu obrazów, ale w każdym z nich ekstrakcja wybranych cech sprawia różnego stopnia problemy. Prostymi algorytmami opisującymi obiekty mogą być filtry wykrywające krawędzie, takie jak Sobel, Laplace czy Canny. Choć te metody są proste i szybko wyznaczają cechy obrazów, są trudne do porównania i bardzo zależne od skali, czy rotacji. Istnieją również bardziej skomplikowane algorytmy wyznaczania inwariantnych cech, ze względu na skalę i obrót. Przykładami takich algorytmów są SIFT i SURF. Oba te algorytmy tworzą wektor deskryptorów dla kluczowych punktów na obrazach, ponieważ, dla każdego z tych punktów wyznaczana jest orientacja, algorytm przestaje być zależny od obrotu obiektów na porównywanych obrazach. W [2] przedstawiono porównanie tych algorytmów. Pokazano, szczególnie, że algorytm SURF jest najszybszy, co jest szczególnie ważne w projektach takich jak ten, gdzie czas wykonywania algorytmu jest jednym z kluczowych elementów. Wpływ na wybór tego algorytmu miało również to, że w używanej przez nas bibliotece do przetwarzania obrazów algorytm ten jest zaimplementowany.

Po wyznaczeniu punktów kluczowych dla aktualnej ramki obrazu, oraz dla wszystkich zapisanych obrazów w bazie, potrzebny jest również algorytm porównujący i łączący w pary odpowiednie deskryptory porównywanych obrazów. Algorytmy klasyfikacyjne to ogromny dział przetwarzania danych. Istnieje wiele rozwiązań, z których każde ma swoje zalety i wady. Na podstawie

---

<sup>1</sup>FLANN - Fast Library for Approximate Nearest Neighbors, Szybka biblioteka do aproksymacji najbliższych sąsiadów.

przykładów w dokumentacji biblioteki do przetwarzania obrazów wybraliśmy metodę przybliżonych najbliższych sąsiadów, która tam jest zaimplementowana jako FLANN.



Rysunek 1: Rozpoznana książka wraz z zaznaczonym jej obszarem

W rozdziale 3 zostanie opisana szczegółowa koncepcja rozwiązania, które algorytmy i w jaki sposób zostały użyte, jakie kryteria zostały wybrane przy wybieraniu najlepszego dopasowania. Zostanie szczegółowo opisana użyta technologia, biblioteka do przetwarzania obrazów oraz biblioteki pomocnicze (GUI i tworzenie bazy danych). W rozdziale 4 pokazane zostaną przykładowe efekty działania programu, oraz podsumowana jakość zaimplementowanych rozwiązań.

### 3. Koncepcja proponowanego rozwiązania

Podstawowym założeniem biblioteki jest tworzenie bazy okładek książek które będą później wyszukiwane na obrazach. Aplikacja pozwala na dodanie okładek aktualnie znajdujących się w klatce filmu. Dla każdej okładki wyliczane są punkty kluczowe wraz z deskryptorami, które między sesjami aplikacji mogą być przechowywane na dysku w plikach XML.

Do obsługi formatu XML użyto bardzo prostej, obiektowej biblioteki TinyXML. Pozwala ona na proste i intuicyjne przeglądanie i tworzenie plików XML w oparciu o jego drzewiastą strukturę.

Baza danych składa się z wielu plików. W głównym pliku bazy są zapisane dane o wszystkich książkach: autor, tytuł, ISBN oraz nazwy plików, w których są zapisane informacje o obrazie. Korzeniem pliku jest tag <books>, w którym znajdują się tagi <book> - po jednym dla książki. Dalej, dla każdej książki informacje tekstowe są zapisane w tagach: <author>, <title> i <isbn>. W kolejnych dwóch tagach: <image\_filename> i <image\_info\_filename> są zapisane informacje o plikach, które zawierają odpowiednio: obraz w formacie oraz informacje o obrazie konieczne do poprawnego rozpoznania w formacie xml zapisywane i odczytywane za pomocą funkcji z OpenCV. Dzięki temu przy kolejnym uruchomieniu programu nie jest konieczne ponowne przetwarzanie wszystkich obrazów w bazie.

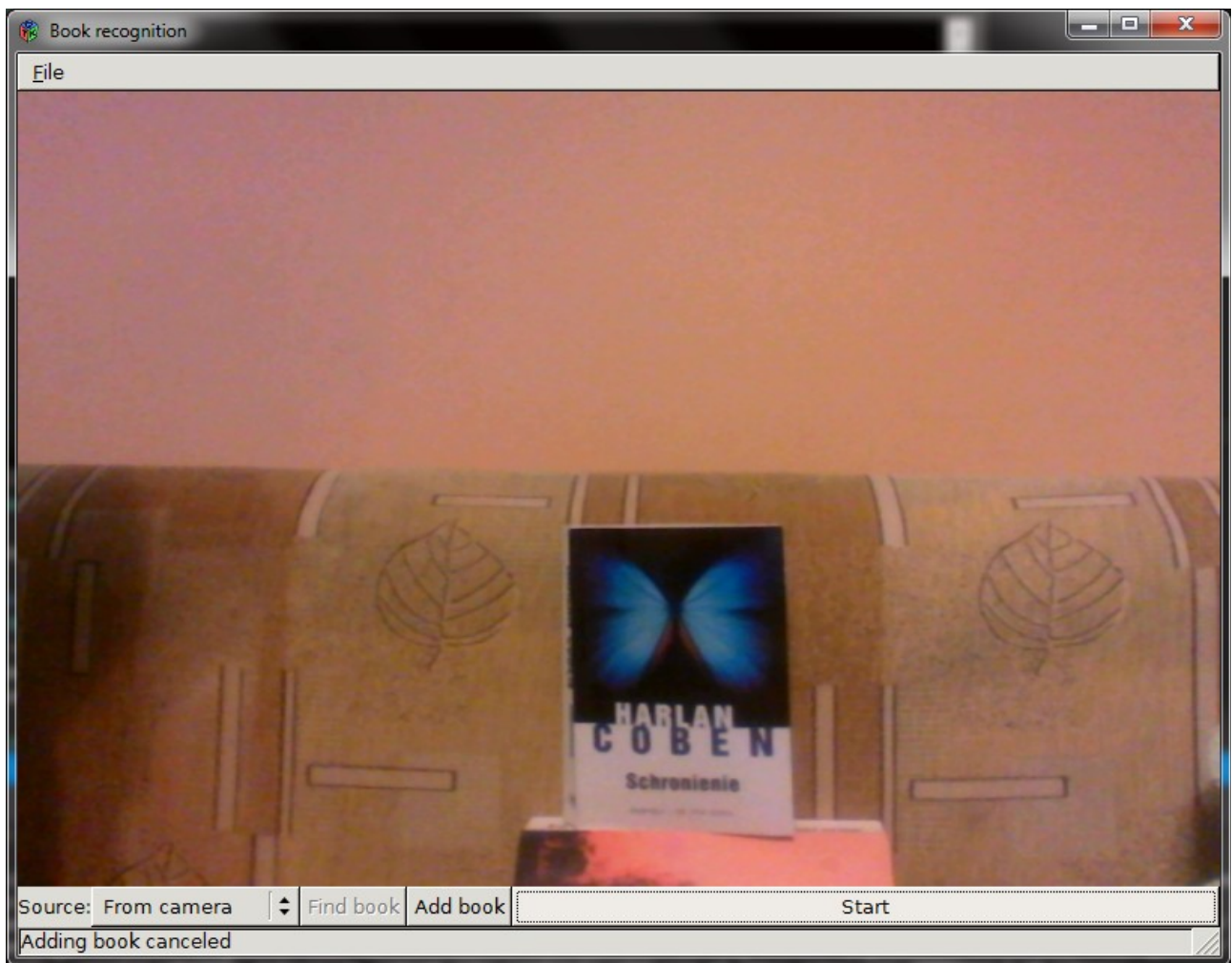
Jako bibliotekę do przetwarzania obrazów wybrano OpenCV. Biblioteka ta zawiera olbrzymią bazę procedur potrzebnych do analizy obrazów. W ostatnim czasie biblioteka przeżywa swój renesans i wiele jej funkcji zostało przepisanych do modułów wykorzystujących do obliczeń karty graficzne (CUDA i OpenCL), co pozwala na znaczne przyspieszenie obliczeń. Biblioteka znajduje się w stanie aktywnego rozwoju.

Dzięki wykorzystaniu tej funkcji wyliczenie cech obrazu jak również ich porównywanie jest bardzo proste. Kryterium wyboru najlepiej dopasowanej książki jest minimalna odległość między deskryptorami obrazu, a deskryptorami okładki. Jeśli ta minimalna odległość najbardziej dopasowanej okładki jest większa niż założony próg, żadna książka nie jest dopasowywana.

Dodatkowym elementem biblioteki jest zapisywanie nowych książek z wybranego fragmentu obrazu. Dokonywana jest korekcja perspektywy w oparciu o 4 wybrane punkty reprezentujące 4 rogi książki.

Z całego programu wydzielono moduł, będący osobną statyczną biblioteką, obejmujący algorytmy wyszukiwania i dopasowania obrazów, na podstawie której zbudowano aplikację do wyszukiwania książek w sekwencjach wideo z pliku lub bezpośrednio z kamery. Program ten został napisany z użyciem biblioteki graficznej GTK. Pozwala ona na szybkie tworzenie prostych interfejsów graficznych i bardzo łatwo integruje się z formatem obrazów biblioteki OpenCV.

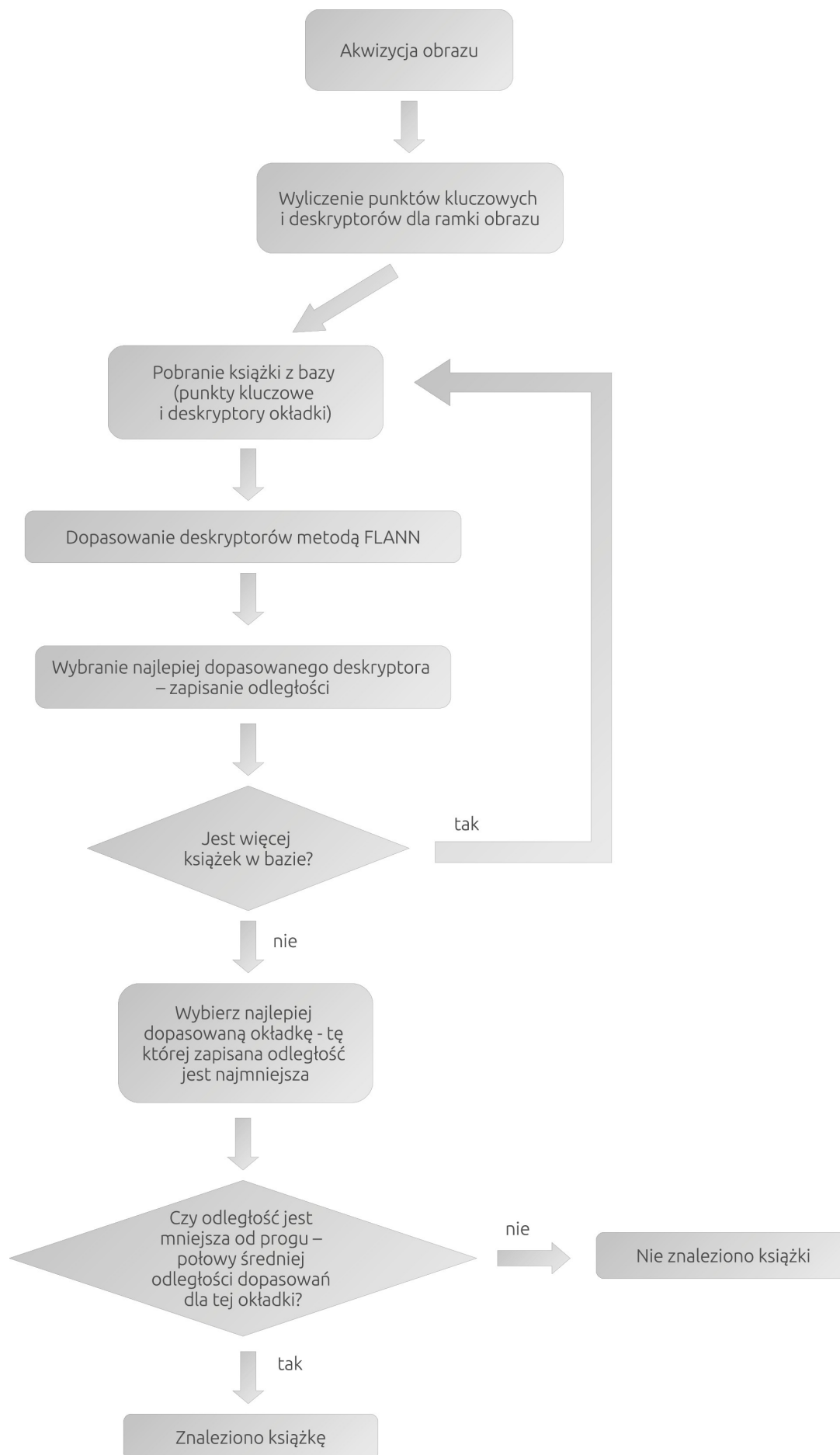
#### 4. Rezultaty i wnioski



*Rysunek 2: Zatrzymana klatka w celu dodania nowej książki*

Rozpoznawania książek udało się zrealizować na poziomie 90%. Błędy są spowodowane zwykle złej jakości obrazem, na którym jest wyszukiwana książką. Są to obrazy nieostre, w ruchu, nieprzedstawiające całej książki. W przypadku, gdy na jednym obrazie znajdują się dwie książki, każda z nich może zostać rozpoznana. Schemat algorytmu wyszukiwania i rozpoznawania książek znajduje się na rysunku 3.

Poniżej jest prezentacja działania aplikacji. Na rysunku 1 widać rozpoznaną książkę. Obszar książki jest zaznaczony jasnozielonym prostokątem. Na rysunkach 2, 4 i 5 widać proces dodawania nowej książki. Na rysunku 4 widać już kilka zaznaczonych punktów.



Rysunek 3: Schemat algorytmu rozpoznawania książki na ramce obrazu

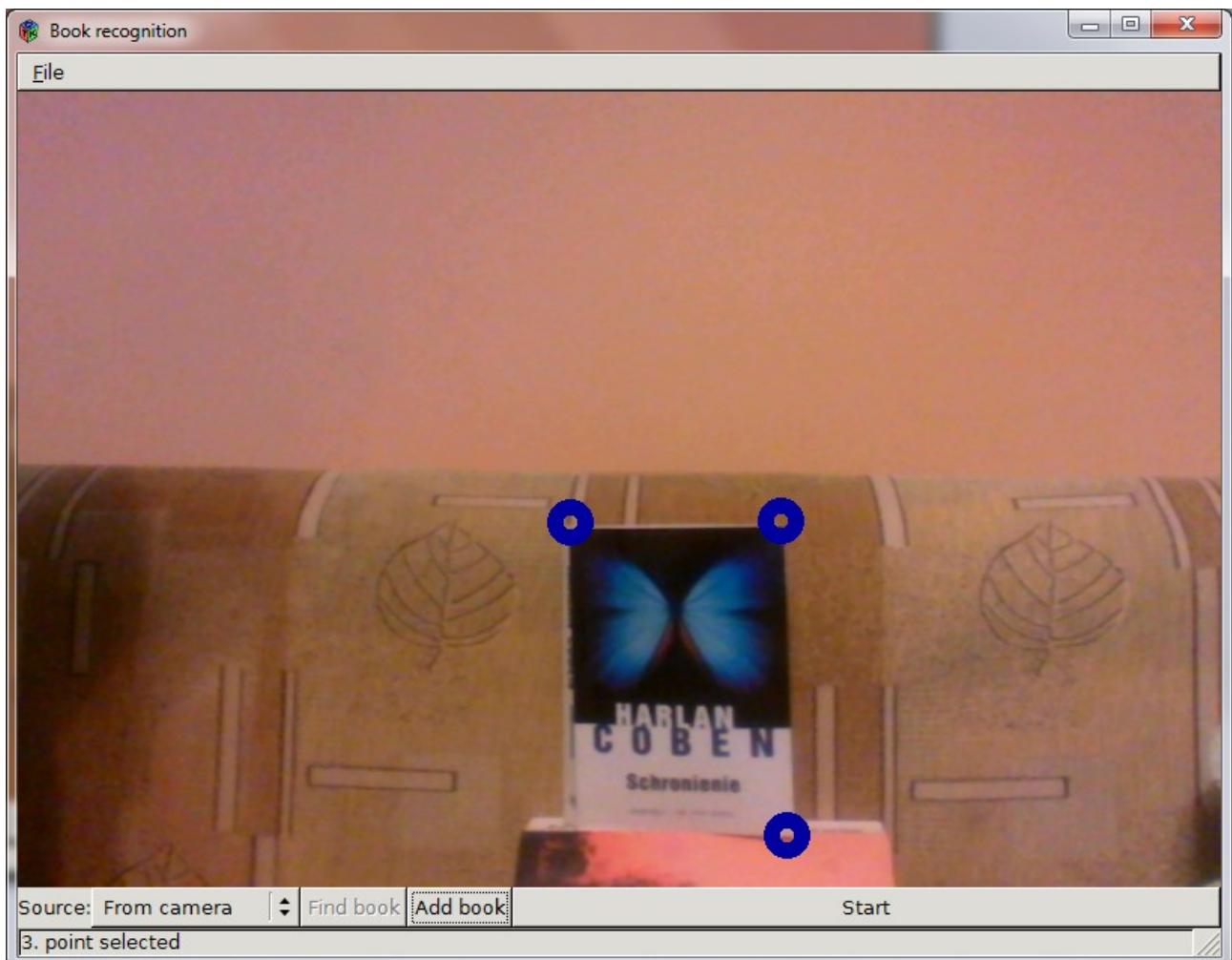


## 5. Podsumowanie

Udało się zaimplementować algorytm wyszukiwania książek po okładce. Jego skuteczność jest dobra. Nie jest on idealny, można znaleźć fragmenty algorytmu, które można poprawić. Aplikacja umożliwia dodawanie nowych książek, obsługę kamery lub wczytywania filmu z pliku, zapis i odczyt bazy.

W dalszych pracach zalecane jest poprawa algorytmu rozpoznawania, ponieważ w tym momencie obliczenia trwają dość długo. Dodatkowo możliwe jest przeniesienie aplikacji na systemy mobilne. W tym celu można przenieść same obliczenia mające na celu rozpoznanie książki do chmury. Dodatkowo można poprawić interfejs aplikacji, zwłaszcza proces dodawania nowej książki, tak aby był bardziej intuicyjny i powodował mniej ewentualnych błędów.

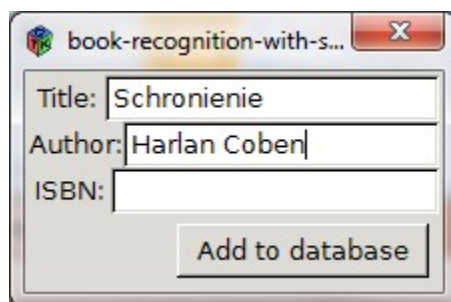
## 6. Literatura



Rysunek 4: Trzy zaznaczone punkty rogów książki

- 1: Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, SURF: Speeded Up Robust Features
- 2: Luo Juan, Oubong Gwun , A Comparison of SIFT, PCA-SIFT and SURF,

3. OpenCV, <http://opencv.org/>
4. Gtkmm, <http://www.gtkmm.org/en/>
5. TinyXML, <http://www.grinninglizard.com/tinyxml/>



*Rysunek 5: Okno dodawania informacji o książce*

## 7. DODATEK A: Opis opracowanych narzędzi i metody postępowania

Zrealizowany program posiada GUI, dzięki czemu jest łatwy w obsłudze. W menu można wczytać, zapisać oraz wyczyścić bazę danych. W głównej części okna widoczny jest obraz z kamery lub wyświetlany jest film. W dolnej części są przyciski pozwalające na działanie aplikacji:

- Przycisk start/stop pozwalający uruchomić i zatrzymać odtwarzanie filmu lub obrazu z kamery,
- przycisk do wyboru źródła odtwarzania (kamera lub plik wideo),
- przycisk do uruchomienia algorytmu rozpoznawania książki,
- przycisk pozwalający dodać książkę widoczną na ekranie do bazy. W tym celu po wciśnięciu przycisku należy wybrać cztery punkty – rogi danej książki oraz uzupełnić dane książki po pojawieniu się odpowiedniego okna. Rogi należy wybierać w następującej kolejności: górny lewy róg, górny prawy róg, dolny prawy róg, dolny lewy róg.

W celu dodania książki do bazy danych należy:

- wybrać plik wideo lub obraz z kamery,
- kliknąć „Start”,
- zatrzymać obraz w momencie, gdy na obrazie będzie widoczna wyraźnie książka klikając „Stop”,
- kliknąć „Add book” a następnie zaznaczyć książkę za pomocą myszy klikając kolejno na punkty: lewy górny róg okładki, prawy górny róg okładki, prawy dolny róg okładki, lewy dolny róg okładki,
- uzupełnić dane o książce: autora, tytuł, ISBN.

W celu rozpoznawania obecnych w bazie książek należy:

- wczytać bazę lub dodać książki



- wybrać plik wideo lub obraz z kamery,
- kliknąć „start”
- w momencie, gdy na obrazie będzie widoczna książką kliknąć „Find book”
- po zakończeniu obliczeń na ekranie pojawią się dane o rozpoznanej książce.

## 8. DODATEK B: Realizacja proponowanego rozwiązania

Program został napisany w standardzie C++11 i jest możliwy do skompilowania i uruchomienia w systemach:

- Windows – za pomocą Visual Studio 2012,
- Linux – za pomocą GCC.

Dokumentacja techniczna z uwagi na swój rozmiar znajduje się na płycie w katalogu DOC.

## 9. DODATEK C. Opis informatyczny procedur

Implementacja została zrealizowana z użyciem kompilatorów Visual Studio 2012 oraz GCC. Zastosowano niektóre elementy standardu C++11, które są zaimplementowane w obu kompilatorach.

Do realizacji użyto bibliotek:

- OpenCV 2.4.2– do realizacji operacji na obrazach,
- TinyXML – do obsługi formatu XML,
- Gtkmm 2.4 wraz z bibliotekami wymaganymi przez Gtkmm – do stworzenia GUI.

Program został podzielony na klasy:

- MainWindow – odpowiedzialna za interakcję z użytkownikiem,
- Database – odpowiedzialna za obsługę plików XML, w których jest zapisana baza książek
- Book – klasa reprezentująca książkę
- Recognizer – klasa odpowiedzialna za rozpoznawanie.

## 10. DODATEK D. Spis zawartości dołączonych nośników (dyskietek, CD ROMu)

Struktura dołączonej płyty CD

- DATA – przykładowa baza danych (plik db.xml wraz z plikami pomocniczymi) oraz film (MOV03427.MPG), na którym są przedstawiane kolejne książki znajdujące się w bazie,
- SRC – źródła programu wraz z plikami dla Visual Studio oraz Cmake, do kompilacji konieczne jest dołączenie bibliotek,
- EXE – plik wykonywalny dla systemu Windows wraz z niezbędnymi plikami dlk,

- DOC - tekst raportu w postaci elektronicznej (MS WORD, OpenOffice oraz PDF) oraz dokumentacja techniczna projektu.