

УДК 621.316.933.3

## Електронна система слідування за положенням об'єкта

*Роздорожнюк Д.М., к.т.н., доц. Кобак М.М.*

*Анотація* – Пропонується розглянути методи з пошуку об'єктів на зображенні, структура пристрою, схема підключення та програмне забезпечення.

*Abstract* - It is proposed to consider methods for finding objects in the image, the structure of the device, the scheme of connection and software.

Ключові слова – OpenCV, пошук об'єктів.

### 1. ВСТУП

Безліч практичних завдань від автоматизації контролю на виробництві до конструювання роботизованих автомобілів безпосередньо пов'язані із завданням пошуку об'єктів на зображенні. Для її вирішення можна застосовувати дві різні стратегії, які залежать від умов зйомки - моделювання фону і моделювання об'єкта.

**Моделювання фону** - цей підхід можна застосовувати якщо камера нерухома, тобто ми маємо фон, який мало змінюється, і таким чином можна побудувати його модель. Всі точки зображення, які істотно відхиляються від моделі фону, вважаємо об'єктами переднього плану. Таким чином можна вирішувати

завдання виявлення і супроводу об'єкта.

**Моделювання об'єкта** - цей підхід більш загальний, дозволяється застосовувати у випадках коли фон постійно і істотно змінюється. На відміну від попереднього випадку, тут нам необхідно знати що саме ми хочемо знайти, тобто необхідно побудувати модель об'єкта, а потім перевірити точки картинки на відповідність цієї моделі.

Іноді умови задачі дозволяють комбінувати обидва підходи, це може істотно поліпшити результати.

Далі будемо розглядати застосування другої стратегії, тобто моделювання об'єкта пошуку.

### 2. ОГЛЯД МЕТОДІВ

**Кольорові фільтри** - якщо об'єкт суттєво виділяється на тлі за кольором, то можна підібрати відповідний фільтр.

**Виділення і аналіз контурів** - якщо ми знаємо, що об'єкт має форму, наприклад, кола, то можна пошукати окружності на зображенні.

Якщо об'єкт на тлі за кольором істотно не виділяється і має складний окрас, то застосування методу кольорових фільтрів не дасть гарних результатів. В цьому випадку можна

спробувати застосувати метод виділення та аналізу контурів. Для цього ми виділяємо межі на зображенні. Межі - це місця різкої зміни градієнта яскравості, їх можна знайти з допомогою методу Canny [3]. Далі ми можемо перевірити виділені лінії - межі щодо відповідності геометричним контурам об'єкта, це можна зробити застосувавши метод Хафа (Hough Transform) [3], наприклад ми можемо пошукати в межах окружності.

**Зіставлення з шаблоном** - у нас є зображення об'єкта, шукаємо в іншому зображенні області збігаються з цим зображенням об'єкта.

Якщо зображення має безліч дрібних деталей то аналіз контурів може бути неефективним. В цьому випадку можна застосувати метод зіставлення з шаблоном (template matching). Він полягає в наступному - беремо картинку з об'єктом і шукаємо на великому зображенні області які збігаються із зображенням об'єкта.

**Робота з особливими точками** - на зображенні з об'єктом шукаємо особливості (наприклад кути), які намагаємося зіставити з такими особливостями на іншому зображенні.

**Метод зіставлення з шаблоном** шукає точні співпадіння точок шаблону з точками зображення. Якщо зображення повернуто чи зміннені розміри щодо параметрів шаблону то цей метод працює погано. Для подолання цих обмежень застосовують методи засновані на так званих - особливих точках. Особлива точка (key point) - це невелика

область, яка істотно виділяється на зображенні. Існує кілька методів визначення таких точок, це можуть бути кути (Harris corner detector) або БЛОБ (blob, крапля), тобто невеликі області однакової яскравості, з досить чіткою межею, що виділяються на загальному тлі.

Для особливої точки обчислюють так званий дескриптор - характеристику особливої точки. Дескриптор обчислюють за заданою околицею особливої точки, як напрямки градієнтів яскравості різних частин цієї околиці. Існує кілька методів розрахунку дескрипторів для особливих точок: SIFT, SURF, ORB і ін. Треба зазначити, що деякі методи розрахунку дескрипторів є запатентованими (наприклад SIFT) і їх комерційне використання обмежене

Особливі точки можна застосовувати для пошуку об'єкта на зображенні. Для цього нам необхідно мати зображення об'єкта який ми розшукуємо і далі виконати наступні дії.

На зображенні з об'єктом шукаємо особливі точки об'єкта і обчислюємо для них дескриптори.

На зображенні який ми аналізуємо теж шукаємо особливі точки і обчислюємо для них дескриптори.

Порівнюємо дескриптори особливих точок об'єкта та дескриптори особливих точок, знайдених на зображенні.

Якщо знайдено достатню кількість відповідностей то помічаємо область з відповідними точками.

**Методи машинного навчання** - навчаємо класифікатор на картинках з об'єктом, деяким способом поділяємо зображення на частини, перевіряємо класифікатором кожен частину на наявність об'єкта.

Метод пошуку об'єктів шляхом порівняння наборів особливих точок має свої недоліки, один з них це погана узагальнююча здатність. Якщо у нас стоїть завдання, наприклад, виділення осіб людей на фото, то по особливих точках наш метод буде шукати одну конкретну фотографію. Ту фотографію, на якій були виділені особливі точки, інші особи будуть виділятися гірше, тому як їм, швидше за все, відповідають інші набори особливих точок.

Результати можуть бути ще гірше якщо змінити ракурс зйомки. Для вирішення цих проблем нам вже необхідні методи машинного навчання і не одна картинка з об'єктом але цілі навчальні набори з сотень (а в деяких випадках - сотень тисяч) різних картинок із зображенням об'єкта в різних умовах.

Якщо розглядати задачу пошуку об'єктів на відео, то її можна розділити на три підзадачі - виявлення, розпізнавання та супроводження.

1. виявлення (detection) - виділяємо області на зображенні, які можуть містити цікаві для нас об'єкти.

2. розпізнавання (recognition) - уточнюємо типи знайдених об'єктів,

3. супроводження (tracking) - локалізація на наступних кадрах розпізнаних об'єктів.

Машинне навчання можна застосовувати, принаймі, для вирішення завдань виявлення і розпізнавання, які зводяться до побудови класифікатора зображень.

Класифікатор зображень складається з двох частин: метод вилучення ознак (feature extractor) і власне класифікатор.

Далі ми розглянемо різні способи побудови класифікатора зображень за допомогою методів вилучення ознак

### **Гістограми спрямованих градієнтів (HOG)**

Гістограма спрямованих градієнтів (Histogram of Oriented Gradients, HOG) [1] - метод вилучення ознак із зображень, який дуже схожий на метод обчислення дескрипторів SIFT [2] для особливих точок, тільки обчислюємо його не для межі особливої точки, але для всього зображення. Загальна схема обчислення HOG виглядає наступним чином. Картинка розділяється на частини (осередки), для кожного осередку будуємо гістограму напрямків градієнта яскравості, далі гістограми осередків нормуються по контрасту і об'єднуються.

### **Ознаки Хаара**

Ознака Хаара (Haar-like features) [8] обчислюється таким чином:

- Вибираємо прямокутну область на зображенні,
- розбиваємо її на кілька суміжних прямокутних частин,
- в кожній частині підсумовуємо яскравість точок,
- після чого обчислюємо різницю між цими сумами.

Ця різниця і буде значенням ознаки.



Рис. 1. Признаки Хаара

На основі ознак Хаара побудований метод Віоли-Джонса, який часто використовують для локалізації зображення осіб на фотографіях. Він об'єднує в собі ознаки Хаара, класифікатор AdaBoost і метод ковзного вікна для поділу картинки на частини

### 3. СТРУКТУРА ПРИСТРОЮ

Тут знадобляться тільки два пристрої:

#### 3.1. Raspberry Pi:

Це остання версія Raspberry Pi (рис. 2). У ній вбудовані Bluetooth і Wi-Fi, раніше доводилося використовувати Wi-Fi свисток в одному з портів USB. Всього в RPI3 40 контактів. З 40 контактів 26 є контактами GPIO.

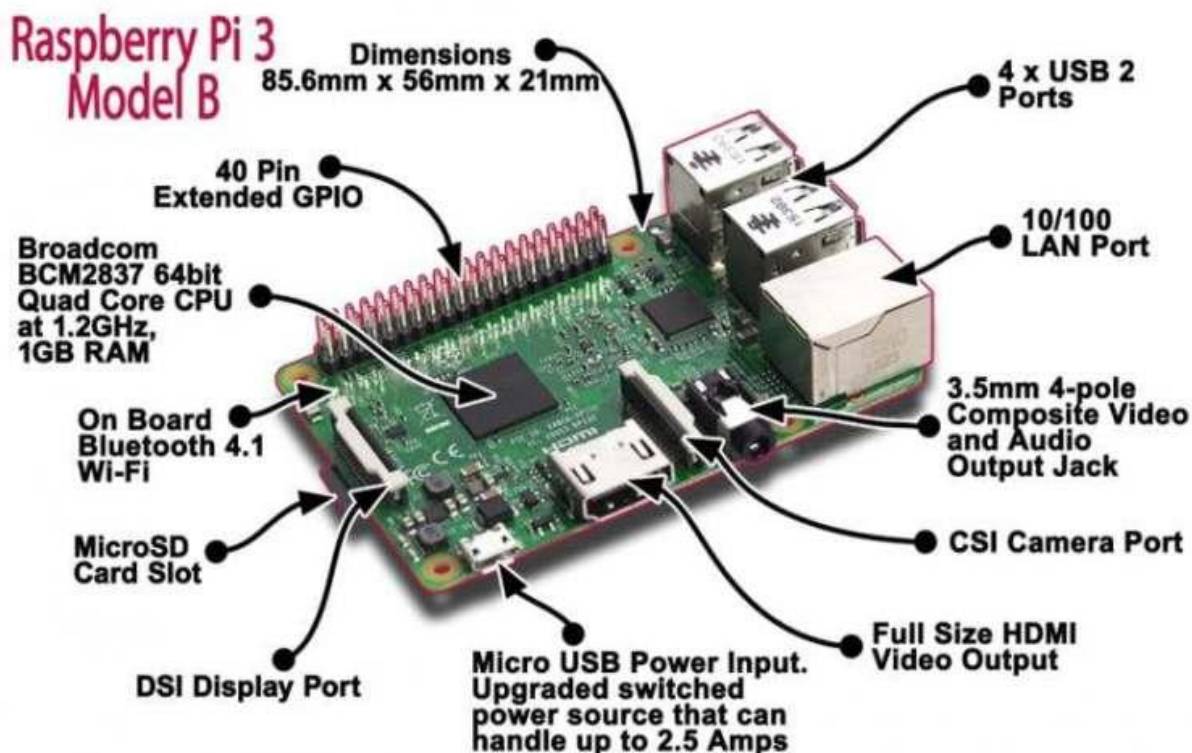


Рис. 2. Плата Raspberry Pi

Є 4 порту USB і 1 слот Ethernet, один порт HDMI, 1 порт аудіовиходу і 1 порт micro usb, а також багато іншого. Є один слот для карт micro sd, в якому ми повинні встановити карту micro sd з рекомендованою операційною системою. Існує два способи взаємодії з вашим Raspberry Pi. Ви можете безпосередньо взаємодіяти через порт HDMI, підключаючи кабель HDMI до VGA, а також клавіатуру і мишу, а також ви можете взаємодіяти з будь-якою системою через SSH (Secure Shell). (Наприклад, через putty ssh.).

### 3.2. Raspberry Pi Camera:

Модуль камери Raspberry Pi (рис. 2, 3) можна використовувати для зйомки відео у високій якості, а також фотографій нерухомих зображень. Його легко використовувати новачкам, але також є багато можливостей які модуль може запропонувати продвинутих користувачам. Існує безліч прикладів онлайн-користувачів, які використовують його для time-lapse, slow-motion відео та інших проектів. Ви також можете використовувати бібліотеки, пов'язані з камерою, для створення ефектів.



Рис. 2. Модуль камери Raspberry Pi

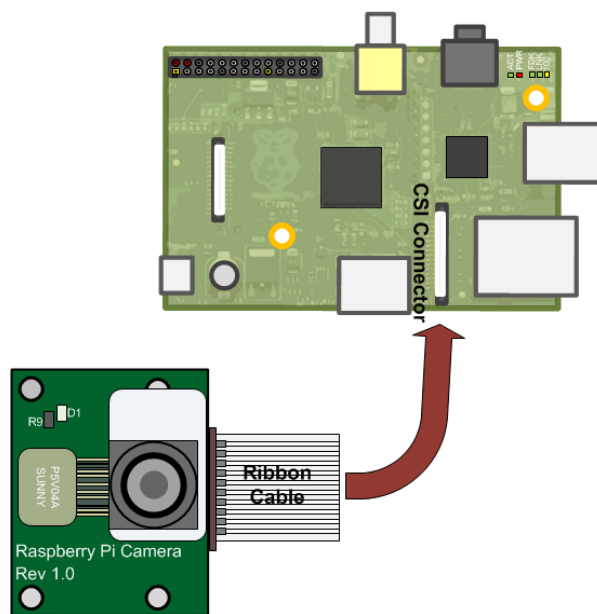


Рис. 3. Схема підключення модуля камери Raspberry Pi

## ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

### OpenCV:

OpenCV (Open Source Computer Vision Library) являє собою бібліотеку програмного забезпечення для комп'ютерного зору і машинного навчання з відкритим вихідним кодом. OpenCV була створена для забезпечення загальної інфраструктури додатків для комп'ютерного зору і прискорення використання сприйняття машини в комерційних продуктах. Будучи ліцензованим BSD продуктом, OpenCV спрощує для бізнесу використати або відредагувати код. У бібліотеці є понад 2500 оптимізованих алгоритмів, які включають в себе повний набір класичних і сучасних алгоритмів комп'ютерного зору і машинного навчання. Ці алгоритми можуть використовуватися для виявлення і розпізнавання осіб, ідентифікації об'єктів, класифікації дій

людини в відео, відстеження руху камери, відстеження рухомих об'єктів і вилучення 3D-моделей об'єктів.

#### 4. ВИСНОВОК

Даний проект можна покращити додавши лазерний далекомір для уточнення відстані до об'єкта, тим самим додаючи третю координату завдяки чому проект працюватиме у тривимірному просторі.

#### Література

1. Е.С.Борисов Детектор объектов для неподвижных камер. - <http://mechanoid.kiev.ua/cv-backgr.html>
2. Е.С.Борисов Детектор объектов для неподвижных камер. - <http://mechanoid.kiev.ua/cv-backgr.html>
3. Е.С.Борисов Детектор объектов для неподвижных камер. - <http://mechanoid.kiev.ua/cv-backgr.html>
4. Антон Конушин Компьютерное зрение (2011). Лекция 3. Простые методы анализа изображений. Сопоставление шаблонов. - <http://www.youtube.com/watch?v=TE99wDbRrUI>
5. OpenCV documentation: Harris Corner Detection - [http://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_feature2d/py\\_features\\_harris/py\\_features\\_harris.html](http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_feature2d/py_features_harris/py_features_harris.html)
6. Wikipedia: Blob\_detection - [http://en.wikipedia.org/wiki/Blob\\_detection](http://en.wikipedia.org/wiki/Blob_detection)
7. Антон Конушин Компьютерное зрение (2011). Лекция 5. Локальные особенности - <http://www.youtube.com/watch?v=vFseUICis-s>
8. Wikipedia: Histogram of oriented gradients - [http://en.wikipedia.org/wiki/Histogram\\_of\\_oriented\\_gradients](http://en.wikipedia.org/wiki/Histogram_of_oriented_gradients)
9. Wikipedia: Scale-invariant feature transform - [http://en.wikipedia.org/wiki/Scale-invariant\\_feature\\_transform](http://en.wikipedia.org/wiki/Scale-invariant_feature_transform)
10. Антон Конушин Компьютерное зрение (2011). Лекция 8. Поиск и локализация объектов. - <http://www.youtube.com/watch?v=I1AiFF6ZkaE>
11. Wikipedia: Haar-like features - [http://en.wikipedia.org/wiki/Haar-like\\_features](http://en.wikipedia.org/wiki/Haar-like_features)
12. P. Viola, M. Jones Robust Real-Time Face Detection - <http://www.vision.caltech.edu/html-files/EE148-2005-Spring/pprs/viola04ijcv.pdf>
13. Е.С.Борисов Бустинг композиции классификаторов - <http://mechanoid.kiev.ua/ml-adaboost.html>