Дигитално процесирање на слика

Тема:

Детекција на регистарски таблички и препознавање на карактери

Драган Велинов – 211245

Опис

Во последните неколку години, светот на компјутерската визија бележи огромни напредоци кои се должат на нови технологии. Заради овие достигнувања, сега е возможно компјутерите со помош на компјутерската визија да ги надминат луѓето во решавање на проблемите од типот на класификација, препознавање на слики, детекција на објекти во слики, препознавање на лица, читање на текст од слика и така натаму.

Со помош на едноствна манипулација на слика, ќе направиме апликација која ќе биде способна да исчита регистарска табличка од било која слика со возило. Потребно е прво да се лоцира регистарската табличка на сликата. За таа цел најпрвин оригиналната слика се конвертира во gray-scale,потоа се прави canny-edge детекција, потоа се наогаат сите правоаголници кој може да се направат со пронајдените темиња и на крај се зема правоаголникот кој најмногу наликува на регистерска табличка. Со модел за препознавање на карактери на слика се отчитува што пишува на табличката.

Користени библиотеки

-OpenCV (Open Source Computer Vision Library) е моќна библиотека првично развиена од Intel за обработка на слики во реално време и задачи за компјутерска визија. Библиотеката нуди над 2500 алгоритми за задачи како што се обработка на слики, откривање објекти, калибрација на слика итн. Отворениот код на OpenCV поттикнува широка употреба во апликации како сателитско мапирање, медицинска обработка на слики, безбедносни системи и роботика. OpenCV е оптимизиран за ефикасност на повеќејадрени процесори, поддржува забрзување на графичкиот процесор за подобрени перформанси и вклучува способности за машинско учење за задачи како препознавање шаблони. Продолжува да се развива со активен развој и широка поддршка од заедницата, што го прави незаменлив за програмерите и истражувачите во компјутерската визија и сродните области.

-Tesseract OCR е систем за оптичко препознавање на знаци (OCR) што прво добива слика како влез преку API интеграција, а потоа ја обработува со алатки како OpenCV за да го подобри квалитетот на сликата, вклучувајќи прилагодување на контрастот и намалување на шумот, што овозможува точно извлекување на текст. Овој систем користи обучени податочни множества и напредни алгоритми, вклучувајќи LSTM невронски мрежи за препознавање на текстуални обрасци во различни големини и стилови на фонтови. Претворениот текст се претставува како битмапа, што овозможува разлика помеѓу текст и графички елементи. Pytesseract, Python обвивка за Tesseract, го олеснува процесот на имплементација и интеракција со функционалностите на Tesseract, поддржувајќи различни формати на слики како JPEG, PNG и TIFF. Исто така, оваа обвивка детектира јазик, ориентација на текст и стил на фонтот, обезбедувајќи информации за граничните полиња на текстуалните региони.

Конвертирање во gray-scale слика

Во овој чекор, алгоритмот ја конвертира влезната слика во слика со “сива скала”. Ова се прави за да се поедностави сликата и да се намали комплексноста на пресметките, бидејќи сликите во “сива скала” содржат само информации за интензитетот (нијанси на сиво), а не информации за бојата. Оваа конверзија ја олеснува обработката на сликата за откривање на рабовите и пронаоѓање на контури.

  
Слика 1.1

  
Слика 2.1

  
Слика 3.1

  
Слика 4.1

  
Слика 5.1

  
Слика 6.1

Canny-edge детекција

Алгоритмот “Canny Edge” детекција се применува на добиената слика за да се откријат рабовите. Рабовите се важни карактеристики кои помагаат во идентификувањето на границите на објектите во сликата. Овој чекор ги истакнува рабовите со идентификување на области на брзи промени на интензитетот.

  
Слика 1.2

  
Слика 2.2

  
Слика 3.2

  
Слика 4.2

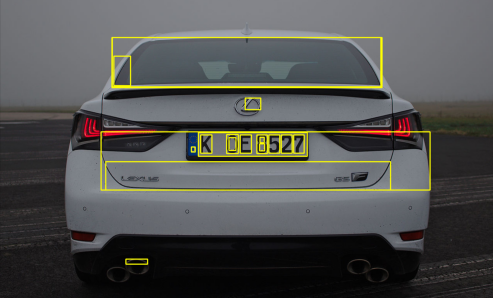
  
Слика 5.2

  
Слика 6.2

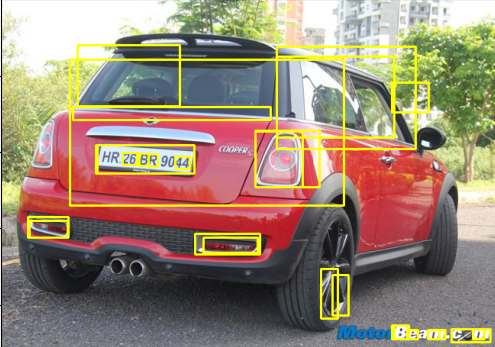
Пронаогање на контури

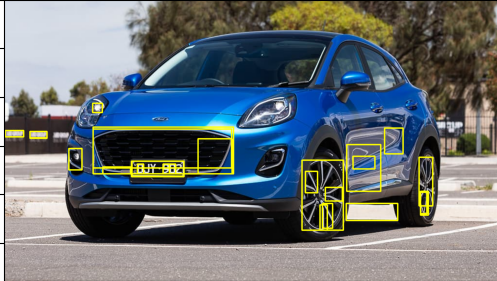
Контурите се криви што ги спојуваат сите континуирани точки долж границата со иста боја или интензитет. Во овој чекор, алгоритмот ги наоѓа сите контури на претходно добиената слика, кој имаат 4 рабови. Контурите се сортираат врз основа на нивната површина, а само најголемите се разгледуваат за понатамошна обработка.

  
Слика 1.3

  
Слика 2.3

  
Слика 3.3

  
Слика 4.3

  
Слика 5.3

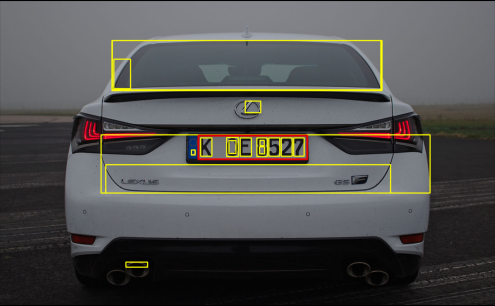
  
Слика 6.3

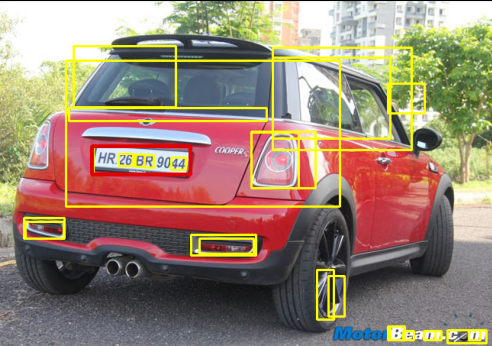
Пронаогање на регистерска табличка

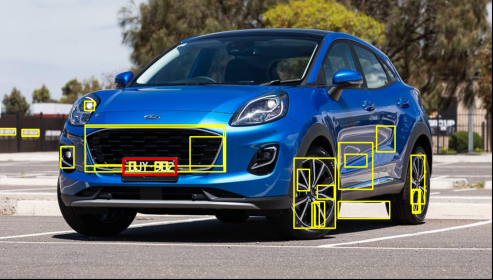
Алгоритмот бара контура со четири агли, бидејќи регистарските таблички обично имаат правоаголна форма. Штом ќе се најде таква контура, се извлекуваат координатите на граничните полиња, а регионот што ја содржи регистарската табличка се означува на сликата.

  
Слика 1.4

  
Слика 2.4

  
Слика 3.4

  
Слика 4.4

  
Слика 5.4

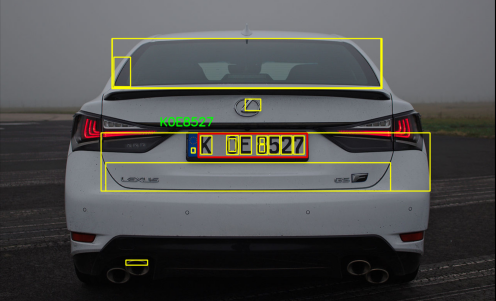
  
Слика 6.4

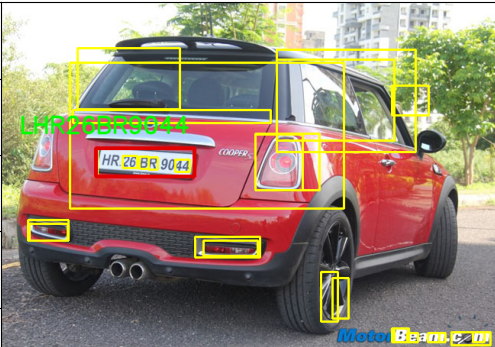
Читање на регистерска табличка

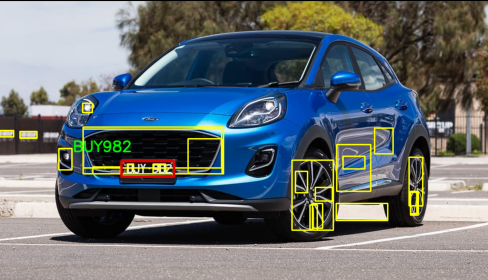
Откако ќе се пронајде регистарската табличка, оптичкото препознавање на знаци (OCR) се користи за читање на текстот од регистарската табличка. Tesseract OCR се применува на изолираниот регион на регистарска табличка за да се извлечат алфанумеричките знаци.

  
Слика 1.5

  
Слика 2.5

  
Слика 3.5

  
Слика 4.5

  
Слика 5.5

  
Слика 6.5