

Automatyczne, wizyjne sortowanie owoców

Bogdan Harasymowicz-Boggio

Instytut Automatyki i Robotyki PW

Streszczenie: Artykuł przedstawia koncepcję urządzenia, które na podstawie obrazu z kamery sortuje owoce pod względem narzuconych kryteriów prawidłowości. Wykorzystanie uniwersalnego systemu wizyjnego do tego zadania jest podejściem nowatorskim, które zdaniem autora umożliwi stworzenie elastycznego automatu, zdolnego do sortowania różnego rodzaju owoców bez konieczności wprowadzania mechanicznych zmian w zależności od owocu. Autor skupia się głównie na systemie wizyjnym i zastosowanych algorytmach przetwarzania obrazu.

Słowa kluczowe: sortowanie owoców, sortowanie wizyjne, automat sortujący

1. Wstęp

Sortowanie, rozumiane jako klasyfikacja i mechaniczne oddzielenie sztuk prawidłowych od wadliwych, jest jednym z ważnych etapów produkcji w przemyśle owocowym. Jednocześnie jest to jeden z etapów najslabiej zautomatyzowanych, czy nawet zmechanizowanych. O ile transport, tłoczenie, czy pakowanie do pojemników odbywa się z wykorzystaniem maszyn i zaawansowanej technologii, sortowanie nadal jest najczęściej dokonywane bezpośrednio przez ludzi nie wyposażonych w żadne wspomagające narzędzia, poza prostym podajnikiem taśmowym. Spowodowane to jest wysokim stopniem skomplikowania czynności percepcyjnych wymaganych w zadaniach sortowania. Choć człowiek jest w stanie wykonać te czynności, proces sortowania przez ludzi przebiega mało wydajnie, gdyż wymaga chwycenia i przeniesienia dłonią każdej odrzuconej sztuki.

Istnieją urządzenia, które pozwalają zautomatyzować sortowanie dla niektórych owoców, lecz charakteryzują się małą elastycznością i budową przystosowaną tylko do konkretnego rodzaju owoców, a także niekiedy bardzo ograniczoną liczbą uwzględnianych cech owocu. Przykładem takiego rozwiązania jest automat Sortex, który do sortowania kawałków malin wykorzystuje wyłącznie czujniki barwy. Nie ma możliwości uwzględnienia kształtu owoców, ani tym bardziej wykorzystania go do innych rodzajów owoców.

Bardziej kompletnym, istniejącym na rynku rozwiązaniem są maszyny sortujące produkowane przez firmę Sorter. W większości tych automatów owoce na podajnikach są umieszczone w specjalnych misach, których kształt jest dostosowany do danego owocu. Pomimo wykorzystania systemów wizyjnych zamiast prostych czujników, rozwiązania tej firmy skupiają się na dużej wydajności sortowania, rezygnując z elastyczności systemu. W średniej wielkości

zakładach, w których są przetwarzane różne rodzaje owoców, zakup oddzielnych automatów do każdego z nich nie jest uzasadnione finansowo.

Celem autora jest opracowanie systemu umożliwiającego automatyczne sortowanie malin w sposób na tyle elastyczny, by możliwe było łatwe jego przystosowanie do innych owoców o zbliżonych rozmiarach. Będzie to możliwe dzięki intensywnemu wykorzystaniu możliwości systemu wizyjnego w połączeniu z uniwersalnymi rozwiązaniami mechanicznymi. Projekt jest rozwijany we współpracy z firmą Madropol.

Projekt opiera się na następujących założeniach:

- 1) Automat będzie sortował głównie owoce zamrożone.
- 2) Priorytetowym owocem jest malina, lecz zarówno konstrukcja, jak i system wizyjny powinny być tak zaprojektowane, by w przyszłości możliwe było ich wykorzystanie do innych owoców o zaokrąglonym kształcie i zbliżonych rozmiarach.
- 3) Ze względu na najwyższą opłacalność finansową, oraz istniejącą technologię komplementarną, celem sortowania jest oddzielenie owoców bez żadnych wad (najwyższej jakości, tzw. klasy „ekstra”) od całej reszty.
- 4) Prawidłowe owoce nie mogą zostać uszkodzone w procesie sortowania. Ograniczenie to nie dotyczy owoców, w których rozpoznano jakąś wadę (odsortów).
- 5) Niewielkie kawałki owoców zostaną odrzucone metodami mechanicznymi przed właściwym sortowaniem.

Dotychczasowe prace nad projektem skupiały się głównie na systemie wizyjnym, który został już prawie ukończony i znajduje się w fazie testowej.

2. Projekt PIKASO

Projektowi (oraz samemu urządzeniu) nadano nazwę PIKASO – Pneumatyczny Inteligentny Kaskadowy Automat Sortujący Owoce. Nazwa ta zawiera najbardziej istotne cechy automatu, które zostaną poniżej krótko omówione.

Urządzenie do sortowania owoców powinno być możliwe jak najbardziej wydajne, niezawodne, łatwe w konserwacji i nie powodować uszkodzeń w sortowanych owocach. Dlatego zostaną zastosowane często spotykane w przemyśle spożywczym dysze powietrzne jako akulatory mechanicznie oddzielające owoce. Stąd automat jest „pneumatyczny”.

Pierwszym problemem konstrukcji, przed którym stanęli projektanci, jest wymóg sprawdzenia „prawidłowości” owocu z każdej jego strony. Rozważane były trzy alternatywne koncepcje:

Prostym, lecz wymagającym sprzętowo rozwiązaniem byłoby badanie owoców podczas spadania (w locie) za pomocą dwóch kamer jednocześnie, lub nadanie im takiej prędkości obrotowej podczas spadku, by ocenić je z każdej strony jedną kamerą. Jednak, ze względu na kruchość owoców mrożonych, należy wykluczyć metody wymagające wszelkiego rodzaju uderzeń owoców, które nie są odsortami.

Drugim możliwym rozważanym rozwiązaniem tego problemu jest zastosowanie takiego podajnika, przez który przechodzi światło, tak, by możliwa była obserwacja owoców od góry i od dołu jednocześnie.

Jedną możliwością realizacji tego pomysłu jest wykorzystanie podajnika z materiału przezroczystego. Jednak w przypadku owoców nie jest to dobre rozwiązanie, gdyż taki podajnik szybko zabarwiłby się sokiem i przestałby pełnić swoją rolę.

Inną możliwością jest zastosowanie siatki – wówczas wystarczyłoby na etapie przetwarzania obrazu uwzględnić moduł usuwający regularną siatkę z obrazu, a następnie dokonać oceny owoców z obydwu stron. Rozwiązanie to jednak zostało odrzucone po konsultacji z ekspertami firmy Madropol, gdyż siatka mogłaby uszkodzić zamrożone owoce.

Ostatecznie więc zaproponowano trzecie, autorskie rozwiązanie, polegające na zastosowaniu podajnika wielostopniowego. Według tej koncepcji występuje jedna kamera, za pomocą której owoce będą śledzone przez system wizyjny i oceniane kilka razy, staczając się po niewielkich stopniach podajnika. Ostateczna klasyfikacja owocu odbędzie się na ostatnim stopniu, gdzie zostanie uznany za prawidłowy wtedy i tylko wtedy, gdy nie wykryto jego wadliwości na żadnym ze stopni. W tym rozumieniu automat jest “kaskadowy”.

Jest to podejście probabilistyczne, gdyż dla danego owocu nie ma absolutnej pewności, że został oceniony ze wszystkich stron, lecz odpowiednia liczba stopni jest w stanie zapewnić dowolnie małe prawdopodobieństwo błędnej klasyfikacji.

Aby oszacować błąd wynikający z zastosowanej metody, rozważmy moduł systemu wizyjnego działający na pojedynczym obrazie, który zawsze poprawnie klasyfikuje prawidłowe owoce i poprawnie klasyfikuje wadliwe owoce wtedy i tylko wtedy, gdy na obrazie jest widoczna jakaś wada. Przyjmijmy, że dla wadliwej maliny prawdopodobieństwo ukazania się wady na pojedynczym obrazie (a zatem prawidłowej klasyfikacji) wynosi 70% (uwzględniając wszystkie rodzaje możliwych nieprawidłowości – zarówno barw, jak kształtu). Przyjmijmy także, że udział wadliwych malin wynosi 50%. Oznaczmy zdarzenia:

A – nieprawidłowa klasyfikacja maliny

B – malina jest wadliwa

C – malina jest prawidłowa

Wówczas:

$$P(A) = P(A/B) \cdot P(B) + P(A/C) \cdot P(C) \quad (1)$$

Dla automatu o N stopniach, proces klasyfikacji powtarzany jest N-krotnie, co (przyjmując założenie, że sprawdzenia te są od siebie niezależne), odpowiada N niezależnym próbom losowym. Zatem prawdopodobieństwo nieprawidłowej klasyfikacji dla automatu wielostopniowego wynosi:

$$P(A_N) = P(A)^N \quad (2)$$

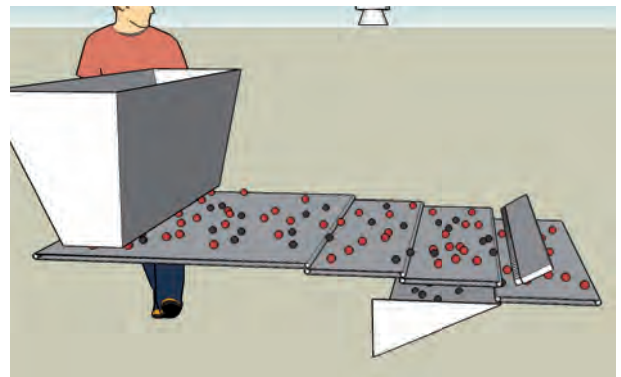
Przyjmując trzy stopnie klasyfikacji, oraz uwzględniając wymienione wcześniej założenia, otrzymujemy:

$$\begin{aligned} P(A) &= P(A/B) \cdot P(B) + P(A/C) \cdot P(C) = \\ &= 0.3 \cdot 0.5 + 0 \cdot 0.5 = 0.15 \end{aligned} \quad (3)$$

$$P(A_N) = P(A)^N = 0.15^3 = 0.0034 = 0.34\% \quad (4)$$

Jak widzimy, błąd jest niewielki, zaś planowany udział wadliwych owoców w sortowanych partiach jest mniejszy niż 50%, co spowoduje dalsze zmniejszenie błędu. Ograniczenie liczby stopni klasyfikacji do 2 przy tych założeniach zwiększyłoby błąd do ponad 2%, co już mogłoby zauważalnie pogorszyć ogólną skuteczność klasyfikacji.

Trzecim z wymagających wyjaśnienia członów nazwy automatu jest “inteligentny”. Ze względu na planowaną elastyczność urządzenia, dostosowanie go do nowego rodzaju owoców lub warunków pracy powinno być jak najprostsze i odbywać się głównie poprzez podanie przez operatora przykładów owoców prawidłowych, bądź wadliwych. Taki sposób uczenia się przez system jest cechą charakterystyczną dla systemów określanymi jako “metody sztucznej inteligencji”.



Rys. 1. Schemat automatu
Fig. 1. Diagram of the machine

3. System wizyjny

Sygnalem wejściowym dla systemu wizyjnego jest ciąg kolejnych obrazów (klatek) przesyłanych przez kolorową kamerę umieszczoną nad podajnikiem taśmowym. Obrazy te mają rozmiar 640 x 480 pikseli (w opinii autora jest to rozsądny kompromis między dobrą widocznością szczegółów, a szybkością obliczeń). Na tych obrazach widoczne są wszystkie trzy stopnie podajnika taśmowego, na którym jadą owoce w kierunku rzędu dyszy. Na taśmie leży tylko jedna warstwa owoców, które wyróżniają się barwą od tła, lecz mogą się ze sobą stykać.

3.1. Oddzielenie od tła

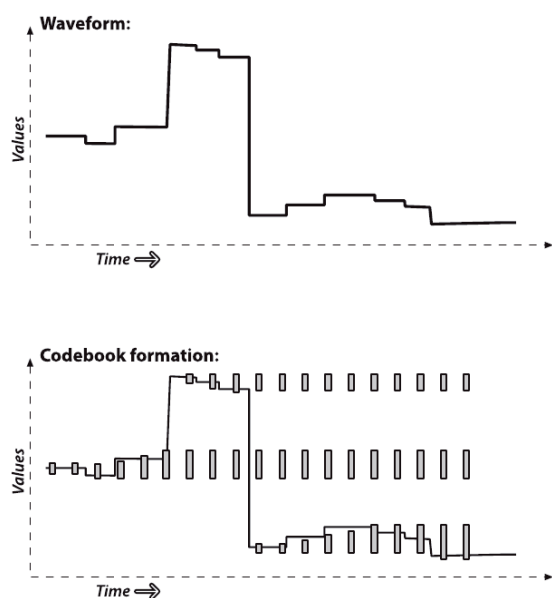
Pierwszym zadaniem systemu wizyjnego jest odrzucenie pikseli należących do tła (widoczne obszary podajnika). Zadanie jest stosunkowo proste, gdyż projektując automat

możemy zastosować taśmę o dowolnej, jednolitej, kontrastującej z owocami barwie, zapewnić stałe warunki oświetleniowe, oraz stałą pozycję kamery względem podajnika.

W systemie wizyjnym PIKASO została zaimplementowana zmodyfikowana wersja algorytmu “codebook segmentation” [2]. W oryginalnej wersji służy on do wykrywania obiektów, które pojawiły się przed znanym, statycznym lub powtarzalnie zmiennym tłem (metoda jest przydatna zwłaszcza w systemach monitorujących, gdzie kamera umieszczona jest na zewnątrz budynku i w tle występują np. drzewa). Dla każdego piksela obrazu tworzony jest oddzielny klasyfikator (zwany “codebookiem”), którego zadanie polega na sprawdzaniu, czy wartość piksela mieści się w jednym z N zapamiętanych w procesie uczenia obszarów (dla przestrzeni RGB są to obszary trójwymiarowe, prostopadłościennie).

Zakładając, że mamy dany ciąg obrazów przedstawiających tło w kolejnych chwilach czasu (ciąg klatek) w przestrzeni RGB, w uproszczeniu proces nauczania (tworzenia modelu tła) można ująć w następujących krokach:

- Dla każdego piksela obrazu sprawdzamy, czy jego wartość (rozumiana jako wektor barwy w RGB) leży w pobliżu jednego z prostopadłościennych obszarów tła zapamiętanych dla współrzędnych tego piksela.
- Jeżeli tak, to w wypadku, gdy wartość ta nie leży wewnątrz obszaru (tylko blisko jego granicy), rozszerzamy obszar w stronę nowej barwy o niewielką, stałą wartość. W ten sposób, jeśli podobna barwa powtórzy się dla danego piksela kilka razy w kolejnych klatkach, obszar rozrośnie się na tyle, że będzie ją zawierał.
- Jeżeli nie, do modelu tła dodany jest nowy, niewielki obszar wyśrodkowany na zaobserwowanej wartości piksela (sytuacja ta ma miejsce dla każdego piksela w pierwszej klatce).



Rys. 2. Ilustracja powstawania modelu tła [2]

Fig. 2. Illustration of the background model creation process

Gdy model tła jest ukończony, oddzielanie od tła polega tylko na sprawdzaniu, czy wartość piksela mieści się

w jednym z zapamiętanych obszarów swojego “codebooka”. Ogólnie można powiedzieć, że algorytm ten stanowi superpozycję wielu dynamicznie powstających, prostych klasyfikatorów progujących podwójnie.

Ponieważ na scenie automatu sortującego owoce podajnik stanowiący tło jest w ruchu, nie ma sensu tworzenie modelu tła dla każdego z pikseli osobno. Jeśli założymy, że oświetlenie podajnika jest równomierne i podajnik jest jednolitego koloru, wystarczy zastosowanie jednego “codebooka”, by stworzyć pełny model tła, ponieważ wszędzie na obrazie możemy się spodziewać tych samych barw i będą one miały takie samo znaczenie z punktu widzenia klasyfikacji, niezależnie od położenia.

Wprowadzona modyfikacja pierwotnego algorytmu “codebook segmentation” polega więc na rozroście pojedynczego codebooka, aktualizowanego wartościami kolejnych pikseli obszaru oznaczonego jako tło, zamiast rozrostu tylu codebooków, ile jest pikseli na obrazie wejściowym, aktualizowanych wartościami przyjmowanymi przez te piksele w kolejnych klatkach. Działanie nauczonego klasyfikatora sprowadza się do porównywania wartości każdego piksela obrazu z granicami obszarów zawartych w “codebooku”.

Implementacja tej metody wraz z operacją morfologicznego otwarcia (następujących po sobie operacji dylatacji i erozji obrazu binarnego) i filtrem medianowym okazała się w pełni spełniać postawione wymagania co do tego etapu przetwarzania obrazu.

3.2. Segmentacja

Kolejnym etapem przetwarzania jest segmentacja, czyli podzielenie obrazu na spójne obszary powiązanych pikseli (w opisywanym systemie segmentacja polega na logiczne powiązanie wszystkich pikseli każdego z owoców). W przypadku, gdy owoc nie styka się z innymi, jest to zadanie proste, gdyż może być rozwiązane metodą rozrostu ziarna na podstawie obrazu uzyskanego w poprzednim etapie przetwarzania. Jednak w przypadku, gdy owoce się stykają, zadanie segmentacji jest znacznie bardziej wymagające.

Oddzielenie owoców metodami mechanicznymi jest możliwe, lecz wprowadzałyby znaczne komplikacje związane z dostosowaniem automatu do sortowania owoców o różnych kształtach i podniosłoby koszt wykonania. Dlatego bardziej korzystne jest większe uproszczenie budowy mechanicznej i przeniesienie części zadań na niematerialną, programową część urządzenia.

Testowane były metody segmentacji oparte na szukaniu krawędzi, progowaniu adaptacyjnym, konwolucji obrazu binarnego, algorytmie watershed [4], oraz złożonych operacjach morfologicznych, lecz zostanie tutaj opisany tylko autorski algorytm, który okazał się dawać najlepszy efekt. Podobnie jak większość zbadanych algorytmów, składa się on z dwóch etapów: znalezienia punktu lub obszaru należącego do każdej z malin (tzw. markera), oraz następnie wykorzystania go do właściwego oddzielenia owoców.

W pierwszym etapie segmentacji markery zostają znalezione za pomocą metody opartej na transformacji Hougha dla okręgów połączonej z silną morfologią. Transformata Hougha dla okręgów jest operacją pozwalającą na znalezienie na obrazie kształtów zbliżonych do okręgów o określonych właściwościach [5]. Ponieważ kształty malin odbiegają (niekiedy znacznie) od kształtu koła, została

zastosowana wstępna operacja wielokrotnego morfologicznego otwarcia (erozji, po której następuje dylatacja) za pomocą elementu strukturalnego o kształcie zbliżonym do okręgu. Operacja ta sprawia, że wszystkie kształty na obrazie zostają wyraźnie “zaokrąglone”, co ułatwia znalezienie markerów za pomocą transformaty Hougha. Rozmiar elementu strukturalnego oraz liczba iteracji otwarcia zostały dobrane eksperymentalnie tak, by uzyskać jak najlepszy efekt, nie tracąc istotnych informacji.

Metoda ta w większości przypadków umożliwia wykrycie wszystkich prawidłowych malin. Segmentacja prawidłowych malin jest priorytetowa, gdyż w wypadku błędnego oddzielenia wadliwych sztuk, i tak zostaną odrzucone przez algorytm klasyfikacji. Skuteczność zaimplementowanej metody znalezienia markerów udało się jeszcze nieco podwyższyć uśredniając wynik działania algorytmu na wielu klatkach (do tego wykorzystano informację o prędkości przesuwu taśmy).

Ze względu na brak odpowiednich narzędzi do oddzielenia od siebie stykających się malin, konieczne było opracowanie w tym celu nowego algorytmu. Zastosowana metoda opiera się na następujących założeniach:

- Cechą charakterystyczną miejsca stykania się malin, ze względu na zaokrąglone kształty owoców, jest występowanie przewężenia obszaru wydzielonego z tła, który tym owocom odpowiada.
- Markery zostały poprawnie odnalezione (występuje dokładnie jeden marker na każdej z malin).
- Granicę między stykającymi się malinami można przybliżyć odcinkiem.

Zakładając poprawność tych założeń, każde dwie stykające się maliny możemy oddzielić odcinkami, które leżą na prostych przebiegających między ich markerami. Aby dopasować odcinki do przewężeń (a zatem do granicy między malinami), należy je dobrać w ten sposób, by były najkrótsze.

Opracowany algorytm dopasowywania odcinków do przewężeń polega na przeszukiwaniu w ograniczonym zakresie w trzech wymiarach: wzdłuż linii łączącej maliny, dla różnych wartości kątowych nachylenia odcinków leżących na prostych przecinających tę linię, oraz wzdłuż tych prostych.

Aby wyjaśnić działanie algorytmu, wprowadzimy pewne oznaczenia:

- MM – odcinek łączący markery
- C – zbiór punktów leżących wzdłuż odcinka MM, naniesionych z pewną określoną rozdzielczością i oddalonych od jego środka o odległość należącą do pewnego ustalonego zakresu
- A – dyskretny zbiór wartości kątowych w zakresie $[0.25\pi - -0.75\pi]$ radianów
- Biały punkt – pixel zaklasyfikowany w pierwszym etapie przetwarzania jako należący do owocu
- Czarny punkt – pixel zaklasyfikowany w pierwszym etapie przetwarzania jako należący do tła (podajnika)
- Kierunki p, q – umowne nazwy przeciwnych zwrotów na odcinkach dopasowywanych do przewężeń lub prostych,

na których te odcinki leżą. Nie jest tu istotne zdefiniowanie poszczególnych zwrotów, a tylko konsekwentne ich rozróżnianie dla danego odcinka.

- e, f – odpowiednio punkt początkowy i końcowy odcinka oddzielającego segmenty
- x – amplituda poszukiwań przewężenia

Dla każdego c ze zbioru C i dla każdego a ze zbioru A znajdujemy prostą l przecinającą odcinek MM pod kątem a w punkcie c. Na tej prostej będziemy szukać odcinka oddzielającego obszar zaklasyfikowany jako owoce (“biały”) na dwie części.

Niech m będzie punktem leżącym na prostej l, oddalonym o odległość x w kierunku p od punktu c, zaś n punktem symetrycznie położonym po drugiej stronie punktu c na tej samej prostej.

Jeżeli m jest biały, podążamy stopniowo prostą l w kierunku p, aż do napotkania punktu czarnego lub brzegowego na obrazie. Gdy taki punkt napotkamy, oznaczamy go jako e.

Jeżeli m jest czarny, podążamy prostą l w kierunku q, aż do napotkania punktu białego lub brzegowego na obrazie. Jeśli znajdujemy taki punkt przed osiągnięciem punktu n, oznaczamy go jako e. W przeciwnym wypadku działanie algorytmu zostaje zakończone, gdyż istnieje zbyt duża przerwa między owocami, by je oddzielać.

Poszukiwanie punktu f przebiega symetrycznie na prostej l względem punktu c, rozpoczynając od punktu n.

Algorytm zapamiętuje najkrótszy odcinek ze wszystkich znalezionych w ten sposób, a następnie nanosi go na binarny obraz owoców oddzielonych od tła w postaci “czarnych” pikseli, oddzielając od siebie stykające się owoce.

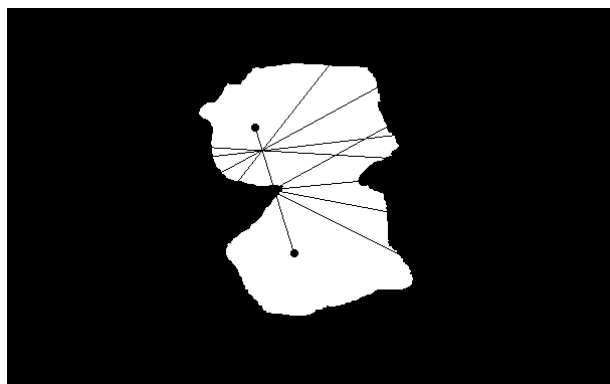
Aby uniknąć dzielenia segmentów tam, gdzie nie jest to konieczne, oddzielone są tylko owoce znajdujące się wystarczająco blisko siebie, oraz dla których nie występują między dwoma markerami markery trzecie w przybliżeniu kolinearnie (sprawia to, że jeśli owoce stykają się w rzędzie, oddzielone są tylko sąsiednie).

Po naniesieniu odcinków oddzielających owoce, nowo powstałe obszary zostają znalezione metodą rozrostu ziarna, zostają zapamiętane ich kontury w postaci łańcuchów punktów, oraz zostaje wyliczony ich geometryczny środek ciężkości w celu dokładniejszego śledzenia obiektów w kolejnym etapie przetwarzania.

Ponieważ zdarzają się sytuacje, kiedy na jednym z owoców (ze względu na nietypowy jego rozmiar lub kształt) nie zostaje znaleziony żaden marker, po zakończeniu opisanego powyżej procesu segmentacji konieczne jest sprawdzenie, czy nie doszło do takiej sytuacji. Dokonano tego poprzez odjęcie od binarnego obrazu owoców obrazu oddzielonych już przez algorytm segmentów, a następnie przeprowadzenie operacji erozji. Jeśli na obrazie zostają jakieś niezerowe (“białe”) obszary, są to niezidentyfikowane owoce, które należy wydzielić z binarnego obrazu początkowego poprzez rozrost ziarna.

3.3. Klasyfikacja

Po ustaleniu granic każdego z owoców na obrazie, należy przystąpić do przyporządkowania ich do jednej z dwu kategorii: owoców prawidłowych, lub wadliwych. Na tym etapie dążymy do minimalizacji błędu klasyfikacji, przy



Rys. 3. Ilustracja działania algorytmu dopasowywania linii do przerwień dla dwóch punktów c na odcinku łączącym markery
Fig. 3. Illustration of the line-fitting algorithm for two c points on the line segment connecting two markers

czym bardziej kosztowne jest przyjęcie wadliwej sztuki jako prawidłowej, niż odrzucenie sztuki prawidłowej. Ważne jest zaprojektowanie systemu tak, by jego uczenie było łatwe dla operatora – najlepiej żeby się odbywało na podstawie kilku przykładów klasyfikacji przez człowieka.

Postawą każdego algorytmu klasyfikacji jest wybór przestrzeni cech, które będą rozpatrywane. W naszym przypadku do dyspozycji mamy dwa rodzaje informacji: informację geometryczną (dotyczącą samego konturu), oraz informację o barwach.

Prostą, lecz bardzo użyteczną w klasyfikacji cechą geometryczną owocu jest jego pole powierzchni na obrazie, gdyż jednym z narzuconych przez klientów kryteriów prawidłowości owocu jest odpowiedni jego rozmiar. Dążymy zatem do odrzucenia zbyt małych owoców (o zbyt małej powierzchni). Cecha ta została uwzględniona w procesie klasyfikacji.

Poza nieprawidłowym rozmiarem, widocznymi na podstawie samych konturów wadami wielu owoców są: nieregularny kształt, wybrakowania i postrzępione brzegi. Prostą i jednocześnie skuteczną cechą wrażliwą na tego typu nieregularności dla zaokrąglonych obiektów jest współczynnik Malinowskiej. Wyraża się on następującym wzorem:

$$R_m = \frac{\text{obwód}}{2 \cdot \sqrt{\pi \cdot \text{pole}}} - 1 \quad (5)$$

Przydatność obydwu tych cech geometrycznych do klasyfikacji malin potwierdzono doświadczalnie.

Co do cech opisujących barwę, po wielu testach algorytmów opartych na histogramach i różnych standardowych, uczących się klasyfikatorach, ostatecznie z nich zrezygnowano na rzecz prostej, autorskiej metody opartej na codebookach, która okazała się najskuteczniejsza.

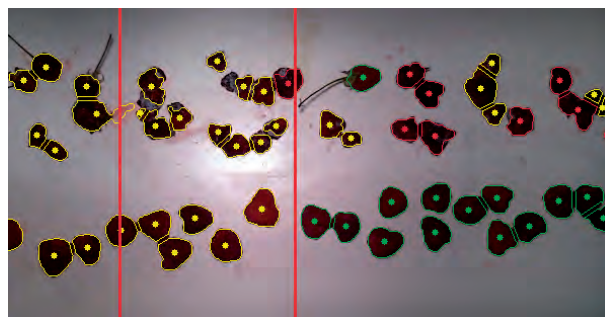
Metoda ta polega na stworzeniu przez użytkownika modelu obszarów prawidłowych poprzez wskazanie prawidłowych owoców (analogicznie do tworzenia modelu podajnika taśmowego w pierwszym etapie przetwarzania). Taki model prawidłowych obszarów jest wykorzystywany do przypisania każdemu pikselowi obrazu kategorii “biały” – czyli prawidłowy obszar owocu, lub “czarny” – niezgodny z modelem obszaru prawidłowych owoców.

Dla każdego obiektu zostają zliczone czarne piksele leżące wewnątrz jego konturu, a następnie liczba ta jest

dzielona przez całkowitą liczbę pikseli wewnątrz konturu (czyli pole powierzchni) w celu uniezależnienia cechy od rozmiaru owocu. Wynik tego działania należy interpretować jako udział wadliwych obszarów w widocznej części owocu i będziemy go nazywać “współczynnikiem wad”. Klasyfikacja ze względu na barwę odbywa się w tej metodzie poprzez porównanie współczynnika wad każdego owocu z ustalonym progiem.

W celu zwiększenia skuteczności klasyfikatora został on uogólniony w taki sposób, że klasyfikacja nie odbywa się tylko na podstawie jednej klatki, lecz wielu. Powiązanie wyników klasyfikacji na kilku klatkach jest możliwe dzięki śledzeniu ruchu każdego owocu z wykorzystaniem wektora prędkości przesuwu taśmy.

W wypadku segmentów, dla których śledzenie nie powiodło się w trakcie klasyfikacji (najczęściej z powodu błędnej segmentacji na niektórych klatkach) nie dysponujemy pełną informacją na temat cech w obszarze oceniania. Traktujemy więc te obszary jako owoce wadliwe, gdyż błędy segmentacji zdarzają się znacznie częściej dla owoców o nieprawidłowych, mniej zaokrąglonych kształtach, które trudniej jest zidentyfikować za pomocą transformaty Hougha. Odrzucenie tych owoców dodatkowo jest uzasadnione wyższym kosztem przyjęcia wadliwej sztuki, niż odrzucenia prawidłowej, co stanowi jedno z założeń projektu. Takie postępowanie pozwala nam skorygować część błędów popełnionych na etapie segmentacji i podnieść ogólną skuteczność sortowania.



Rys. 4. Przykład klasyfikacji opisaną metodą. Między czerwonymi liniami znajduje się obszar oceniania owoców. Mimo iż część odsortów została na którejś wcześniejszej klatce błędnie posegmentowana, ostatecznie błąd decyzji został popełniony tylko dla jednego owocu

Fig. 4. An example of classification using the described method. Between the red lines we see the classification area. Although some of the fruit has been incorrectly segmented, only one final decision error was made

4. Skuteczność systemu wizyjnego

Dokładny pomiar ogólnej skuteczności segmentacji i klasyfikacji owoców w docelowych warunkach jeszcze nie był możliwy, gdyż wciąż trwają prace nad systemem oświetlenia. Dokonano jednak wstępnego oszacowania skuteczności systemu wizyjnego na podstawie dostępnego materiału filmowego, nagranych w warunkach prowizorycznych - wynosi ona powyżej 90%.

Wadliwe owoce, ze względu na nieregularny kształt, czasami nie zostają poprawnie oddzielone na etapie segmentacji, lecz zakładając stosunkowo dużą rozdzielczość rozmieszczenia aktuatorów pneumatycznych, odrzucenie

ich podczas sortowania nie będzie problemem. Dla owoców prawidłowych (których dokładne granice muszą być znane w celu rozpoznania prawidłowości ich kształtu) błędy segmentacji są rzadkie.

5. Podsumowanie

Dotychczas wykonana praca nad projektem PIKASO oraz przeprowadzone badania pokazują, że systemy wizyjne mogą być skutecznie wykorzystywane w zadaniach sortowania owoców, co pozwoli na zwiększenie uniwersalności urządzeń, które te zadania wykonują.

Kolejnym etapem projektu jest stworzenie mechanicznego prototypu automatu sortującego oraz udoskonalenie i rozszerzenie systemu wizyjnego tak, by uwzględniał kilka stopni klasyfikacji oraz był zdolny sortować kilka różnych rodzajów owoców.

Gdy przedstawiona w tym artykule koncepcja automatu sortującego zostanie sprawdzona w praktyce na prototypie, możliwe będzie skonstruowanie docelowego automatu, który ma być wykorzystany do sortowania owoców w firmie Madropol. Optymistyczny plan zakłada ukończenie prac nad projektem przed najbliższym sezonem zbioru malin.

Bibliografia

1. T. Acharya, A. Ray (2005) *Image Processing: Principles and Applications* Wiley.
2. Gary Bradski, Adrian Kaehler (2008) *Learning OpenCV*, O'Reilly.
3. J. Serra (1983) *Image Analysis and Mathematical Morphology* Academic Press.
4. F. Meyer (1992) *Color image segmentation, Proceedings of the International Conference on Image Processing and Its Applications* s. 303-306.

5. C. Kimme, D. H. Ballard, J. Sklansky (1975) *Finding circles by an array of accumulators, Communications of the Association for Computing Machinery* 18 s. 120-122. ■

Automatic, visual fruit sorting

Abstract: Abstract: The article presents the idea of a machine which can sort fruit based on the image stream of a color camera. Using a universal computer vision system for this task is a novel conception, which the author believes can allow to build an automaton able to sort different kinds of fruit without mechanical changes depending on the fruit. The main focus of the article is the vision system and the implemented image processing algorithms.

Keywords: fruit sorting, visual sorting, sorting machine

Bogdan Harasymowicz-Boggio

Jest studentem IV roku robotyki na wydziale Mechatroniki Politechniki Warszawskiej. Jego zainteresowania naukowe dotyczą sztucznej inteligencji, widzenia maszynowego i robotyki mobilnej, szczególnie związanej z uczeniem maszynowym. Bierze udział m.in. w projekcie Robota Kuriera powstającego na Wydziale Mechatroniki Politechniki Warszawskiej.

e-mail: mysticdrow@gmail.com

