System automatycznych pomiarów rynometrycznych (4)

Tomasz Kuśmierczyk

Studenckie Koło Naukowe Cybernetyki, Politechnika Warszawska

Streszczenie: Celem projektowanego systemu jest analiza i rozpoznawanie obrazów trójwymiarowych twarzy. Wykorzystując dostępne metody analizy i narzędzia algorytmiczne dąży się do pozyskania, z danych pochodzących ze skanerów 3D, informacji dotyczących wymiarów nosa. Artykuł przybliża aspekty stosowania jednego z głównych podejść do zagadnienia analizy obrazów 3D, tj. deskryptorów punktów. Przybliżono istniejące rozwiązania. Rozważono różne podejścia do wyboru punktów sąsiednich. Omówiono dobór skali i skwantowania. Wprowadzono odległość między deskryptorami. Pokazano też, jak zastosować deskryptory w rozpoznawaniu obrazów.

Słowa kluczowe: antropometria, deskryptor, rozpoznawanie kształtu, rynometria, analiza twarzy

W poprzednim artykule cyklu dokonano krótkiego wprowadzenia w tematykę metod analizy obrazów trójwymiarowych. Wypunktowano i opisano podstawowe podejścia. Wprowadzono deskryptory punktów. W tej części temat deskryptorów zostanie rozwinięty i pogłębiony.

Przypomnijmy, że deskryptor to pewna struktura charakteryzująca pojedynczy punkt (dalej oznaczany przez p) na powierzchni analizowanego obrazu. W omawianych przypadkach strukturą tą jest dwuwymiarowy histogram położeń punktów sąsiednich (ozn. q) względem punktu centralnego p, dla którego obliczany jest deskryptor. Dotychczas omówiono cztery typy deskryptorów:

- *spin image* (obraz obrotu)
- *local surface patch* (''łatka'' lokalnej powierzchni)
- *shape context* (kontekst kształtu)
- *local shape map* (mapa lokalnego kształtu).

Teraz zostaną omówione praktyczne aspekty ich użycia. W odniesieniu do istniejących rozwiązań pokazane zostaną problemy konieczne do rozwiązania przed użyciem tej techniki w tworzonym systemie.

1. Użycie deskryptorów w istniejących rozwiązaniach

Istnieją dwa podejścia do użycia deskryptorów punktów:

1) Zastosowanie polegające na klasyfikacji pojedynczych punktów. Stosując metody sztucznej inteligencji, próbuje się odnajdywać punkty spełniające dane kryterium.

Takie podejście zastosowano np. w [1]. Podjęto próbę odnalezienia trzech punktów charakterystycznych twarzy: wierzchołka nosa (prn) i kącików wewnętrznych oczu (en). W tym celu klasyfikowano obrazy obrotu punktów na twarzy za pomocą SVM. Odnalezione przybliżone położenia punktów stosowano następnie do normalizacji ułożenia twarzy.

Wadami takiego podejścia są:

- Potrzeba przygotowania odpowiednich rozmiarów i zbioru uczącego o wysokiej jakości.
- Bardzo wysoki koszt obliczeniowy. Wygenerowanie obrazów obrotów (a następnie klasyfikacja) dla wszystkich punktów twarzy jest niezwykle czasochłonne. Redukcję czasu umożliwia wstępna selekcja regionów, dla których generowane będą deskryptory. Zazwyczaj selekcję taką przeprowadza się na podstawie informacji o krzywiźnie obszarów (co jest dość kosztowne).
- Wrażliwość na błędy – pojedyncze niepoprawne punkty danych mogą mieć charakterystyki bardzo podobne do tych poszukiwanych.

2) Zastosowanie, w którym dopasowuje/przyporządkowuje się (ang. *match*) deskryptory zbudowane na podzbiorze punktów analizowanego obiektu do analogicznego podzbioru wzorca. Traktując twarz jako przybliżoną instancję wzorca, wyszukiwanie punktów na twarzy można potraktować w kategoriach problemów znajdowania odpowiedniości (ang. *correspondence problem*) między punktami we wzorcu i testowanym obiekcie/scenie. Deskryptory nie muszą mieć identycznych charakterystyk i nie muszą być przypisane do dokładnie odpowiadających sobie punktów, by zostać do siebie dopasowane [2]. Jest możliwe np. dopasowanie wierzchołka nosa z jednego obrazu twarzy do punktu położonego lekko z boku wierzchołka w drugim obrazie. Należy jednak pamiętać, że problem ma swoją specyfikę: część punktów może nie istnieć, a mogą istnieć regiony nieuwzględnione we wzorcu. Opisany sposób użycia deskryptorów znalazł liczne zastosowania, m.in.:

- Rozpoznawanie obiektów dowolnego kształtu (ang. *free-form objects*). W [3, 4, 5] wykorzystano (ze średnią skutecznością 99,37 % [MNIST]) *shape contexts* do rozpoznawania pojedynczych liter pisma odręcznego. Dopasowywano deskryptory z analizowanych symboli do deskryptorów wygenerowanych dla liter wzorcowych. Następnie wybierano to dopasowanie, którego odległość (sumaryczny koszt) była najmniejsza.

  Podobne podejście, ale do rozpoznawania twarzy z wykorzystywaniem obrazów obrotu, zastosowano w [6] oraz z wykorzystanie sygnatur punktów w [2]. W obu przypadkach bibliotekę stanowił zbiór obrazów 3D twarzy osób.

- Lokalizacja obiektów w scenie. Tutaj użyto obrazów obrotu do poszukiwania obiektów w scenie 3D [7, 8]. Dla kolejnych punktów generowano deskryptory, które następnie (po redukcji wymiarowości za pomocą PCA) dopasowywano do deskryptorów obiektów z biblioteki. W [22] do podobnego celu użyto *local surface patches*.

2. Punkty sąsiednie

Budując histogram rozmieszczenia punktów wokół p zadecydować należy, które z punktów q obrazu 3D zostaną uwzględnione w obliczeniach. W literaturze funkcjonuje kilka podejść.

W obrazach obrotu standardowo brane pod uwagę są te punkty, w których normalna do powierzchni nie różni się znacznie od normalnej w punkcie p:

$$a\cos\left(\overrightarrow{n_p}\overrightarrow{n_q}\right) \le A_s,$$

gdzie A_s – stała.

Wadą takiego podejścia jest konieczność obliczenia wektorów normalnych we wszystkich punktach obrazu, co jest niezwykle czasochłonne.

Innym podejściem jest uwzględnienie wszystkich punktów sceny przy założeniu, że ostatni z przedziałów jest nieskończony. Możliwe jest też ograniczenie punktów wpływających na histogram tylko do tych, które położone są w sąsiedztwie – odległości mniejszej niż r od punktu p:

$$||q - p|| < r.$$

Podejście takie zastosowano w *local shape map* i *point signature*.

3. Skala i skwantowanie

Istotnymi parametrami, które rozważyć należy przy budowie deskryptorów wykorzystujących histogramy są typ skali i jej skwantowanie.

Jak wyjaśniono wcześniej, pierwszym krokiem obliczeń deskryptorów jest przyporządkowanie każdemu punktowi w sąsiedztwie (ozn. q) pary liczb: jego nowych współrzędnych względem p. Do opisu położenia w przestrzeni 3D stosowane są dwie współrzędne, więc nie jest wykluczone, że te same wartości przyporządkowane zostaną do kilku punktów położonych w zupełnie różnych miejscach, np. w obrazach obrotów wszystkie punkty położone na okręgu o środku na prostej wyznaczonej przez wektor normalny i równoległym do płaszczyzny stycznej mają te same współrzędne. W drugim kroku budowany jest dwuwymiarowy histogram rozkładu punktów w nowych współrzędnych. Dla każdego przedziału współrzędnych („kubełka") zliczone zostają wszystkie należące do niego punkty.

Kwestią, którą należy rozważyć jest sposób podziału przestrzeni w nowych współrzędnych. Należy zadecydować, na ile przedziałów zostanie podzielona każda z osi (jaka będzie rozdzielczość) oraz jak zostaną one rozmieszczone (typ skali). Im większa rozdzielczość skali, tym większy wpływ na wartości histogramu mają pojedyncze punkty. Zwiększana jest ilość informacji, jaką niesie ze sobą deskryptor. Tym samym zwiększa się wrażliwość na błędy i wpływ efektu próbkowania powierzchni przy skanowaniu obiektów [7]. Bardzo istotne jest zbalansowanie wpływu tych czynników. W obrazach obrotu rozdzielczość uzależniona została od rozdzielczości całego modelu. W *shape contexts* skwantowanie przyjęto eksperymentalnie.

Oryginalnie dla obrazów obrotów zastosowano skalę liniową, tj. kolejne przedziały mają taką samą szerokość. Natomiast w *shape contexts* dla kątów zastosowano skalę liniową, a dla odległości logarytmiczną. Użycie skali logarytmicznej różnicuje wpływ punktów q położonych w różnych odległościach od punktu p. Obszary w mniejszej odległości reprezentowane są dokładniej, przez co ich późniejszy wpływ przy porównywaniu i dopasowywaniu deskryptorów jest większy [3].

4. Odległość między dwoma deskryptorami

Dla porównania i oceny kosztów dopasowania dwóch deskryptorów wprowadza się pojęcie odległości między nimi. Odległość jest miarą podobieństwa między deskryptorami. Im jest ona mniejsza, tym są one bardziej do siebie podobne. W większości opracowań stosuje się jedną z dwóch miar:

1) odległość stosowana w [3–5] do dopasowywania *shape contexts*, a także w [9] przy przyporządkowywaniu *local surface patch*. Dalej umownie nazywana ona będzie *shape context measure* (SCM):

$$c_1(g,h) = \frac{1}{2}\sum_{k=1}^{K}\frac{\left(g(k)-h(k)\right)^2}{g(k)+h(k)}$$

W przypadku zastosowania SCM histogramy są znormalizowane (tj. suma zawartości „kubełków" wynosi 1).

2) odległość wykorzystująca korelację liniową Pearsona, stosowana w [2, 6–8]. Wersja o wartościach zwracanych zgodnych z *shape context measure*:

$$c_2(g,h) = \frac{1}{2}\left(1 - \frac{\mathrm{cov}(g,h)}{\sqrt{\mathrm{var}(g)\,\mathrm{var}(h)}}\right)$$

gdzie:

- g, h – deskryptory (histogramy), między którymi liczona jest odległość; oryginalnie są one dwuwymiarowe, ale bez wpływu na zachowanie algorytmów można je traktować jako jednowymiarowe (np. układając kolejne wiersze obok siebie)
- k – numer kolejnego „kubełka" histogramu
- K – liczba „kubełków'' histogramu (dla obu histogramów jest ona taka sama)
- *var/cov* – estymator wariancji/kowariancji (kolejne wartości „kubełków" traktowane są jako próby losowe)
- $c \in [0, 1]$ – koszt dopasowania między dwoma histogramami; jeśli $c=0$, deskryptory opisują zupełnie różne regiony, jeśli $c=1$ – deskryptory są identyczne.

5. Filtrowanie deskryptorów

Zaobserwowano [10], że przy budowie deskryptorów wiele „kubełków" zawiera zera. Jest to spowodowane faktem, że punkty obrazu w bardzo małym stopniu (lub wcale) pokrywają przedziały wartości im odpowiadające. Założono, że zera pogarszają jakość dopasowania deskryptorów. W celu redukcji tego efektu wprowadzono proste filtrowanie: „kubełki" zerowe nie są uwzględniane przy obliczaniu odległości między deskryptorami. Możliwe są dwa podejścia:

- filtrowanie AND – dwa odpowiadające sobie „kubełki" $g(k)$ i $h(k)$ brane są pod uwagę przy dopasowywaniu tylko, gdy oba są różne od zera: $g(k)>0$ i $h(k)>0$;
- filtrowanie OR – $g(k)$ i $h(k)$ brane są pod uwagę przy dopasowywaniu, gdy co najmniej jeden z nich jest różny od zera: $g(k)>0$ lub $h(k)>0$.

Regułę, w której wszystkie elementy histogramu brane są pod uwagę oznaczono jako NONE.

Dodatkową korzyścią z wykorzystania filtrów, o której nie wspomniano w [10] jest ich zdolność do usuwania części danych, które nie powinny być uwzględnianie przy dopasowywaniu. Możliwa jest sytuacja, gdy we wzorcu w pewnym regionie nie ma punktów, a układ sceny powoduje, że w analogicznym regionie wyszukiwanego obiektu są punkty będące np. częścią innego obiektu. Stosując filtr AND uniknie się wpływu tych punktów przy dopasowywaniu deskryptorów.

6. Dopasowywanie punktów i algorytm węgierski

Wśród metod dopasowywania (przyporządkowywania, ang. *match*) deskryptorów (i odpowiadających im punktów) na największą uwagę zasługuje podejście zastosowane w połączeniu z *shape contexts*. Założonym celem jest znalezienie takiego dopasowania między parami deskryptorów (jeden z wzorca i jeden z analizowanego obrazu), które zminimalizowałoby sumaryczny koszt:

$$\min_U C = \sum_{(g,h)\in U} c\left(g,h\right) \qquad (*)$$

gdzie przez U rozumie się zbiór dopasowań (transwersalę) między deskryptorami: $g\in G$, $h\in H$. Liczność U jest równa mniejszej z liczności zbiorów deskryptorów wzorca i analizowanego obrazu:

$$|U| = \min(|G|, |H|)$$

gdzie G i H to odpowiednio zbiory deskryptorów z wzorca i obrazu. Oznacza to, że jeśli jeden z tych zbiorów jest liczniejszy to dopasowanych zostanie tylko tyle deskryptorów, ile jest w drugim zbiorze.

W pozostałych podejściach stosuje się różne heurystyki, które zwykle przyspieszają czas obliczeń, jednak w wielu przypadkach powodują, że uzyskane rezultaty są dalekie od optymalnych. Kryterium (*) wymusza sposób dopasowania, który nie jest osiągalny w innych przypadkach. Możliwa jest sytuacja, kiedy przyporządkowywane do siebie są deskryptory, które są od siebie bardzo odległe, ale w łącznej optymalizacji powodują najmniejszy koszt. Dodatkowo, czasochłonność procedury może zostać zbalansowana zmniejszeniem liczby deskryptorów.

Dopasowanie punktów między wzorcem, a analizowanym obiektem (ang. *assignment problem*) w sposób spełniający kryterium (*) jest możliwe w czasie wielomianowym. Najpopularniejszą z metod jest algorytm węgierski z 1955 r. Jego złożoność czasowa w wersji oryginalnej to $O(M^4)$, jednakże Edmonds i Karp oraz niezależnie Tomizawa pokazali, że możliwe jest jego przyspieszenie do $O(M^3)$, gdzie M – liczność większej z grup dopasowywanych elementów. Algorytm w najpopularniejszych implementacjach operuje na macierzach. Wiersze odpowiadają punktom wzorca, kolumny – punktom analizowanej sceny. Na przecięciu umiejscawiany jest koszt przyporządkowania.

Wynikiem działania algorytmu węgierskiego jest macierz reprezentująca dopasowanie między punktami wzorca (wiersz), a obrazu testowanego (kolumna). Macierz zawiera maksymalnie jedną wartość 1 w każdej kolumnie i maksymalnie jedną wartość 1 w każdym wierszu, co oznacza jednoznaczne przypisanie.

Oprócz czasu przyporządkowania dwóch zbiorów G i H, o licznościach odpowiednio: $M_G=|G|$ i $M_H=|H|$, przy ocenie złożoności metody uwzględnić należy też czas przygotowania macierzy kosztów. W pierwszym kroku obliczane są deskryptory dla dwóch obrazów o liczbie punktów odpowiednio N_G i N_H. Koszty wynoszą $O(N_G M_G)$ i $O(N_H M_H)$. Następnie obliczane są odległości między każdymi dwoma deskryptorami: $O(M_H M_G K)$, gdzie K – liczba „kubełków". Dopasowywanie jest ostatnim krokiem.

7. Strategie wyboru podzbioru punktów

Przy zastosowaniu deskryptorów do dopasowywania (ang. *matching*) obiektu (podzbioru punktów poszukiwanego w scenie na obrazie 3D) do wzorca, kwestią podstawową jest wybór punktów, dla których zostaną one obliczone. Problem ten można rozważać niezależnie dla obiektu i wzorca. W literaturze występuje kilka sposobów rozwiązań tego problemu:

1) Generacja deskryptorów dla wszystkich punktów obrazu. Metodę taką zastosowano do sygnatur punktów (ang. *point signatures*) i obrazów obrotu w [7, 11].

Wadą tej metody jest ogromny nakład obliczeniowy zarówno przy generacji deskryptorów, jak i później przy dopasowywaniu grup deskryptorów z wzorca i z obiektu. Zaletą jest lepsza jakość dopasowywania: sumaryczny koszt dopasowania dwóch identycznych obrazów 3D jest równy 0. Omijany jest też (opisany dalej) problem względnego przesunięcia punktów.

2) Generacja deskryptorów dla punktów, w których znajdują się ekstrema lokalne wartości indeksu kształtu (ang. *shape index*). Punkty takie wydają się najlepiej reprezentować charakterystykę danego regionu. Podejście takie pojawia się w związku z LSP [9, 12].

Problemem jest wysoki koszt obliczenia wartości indeksu kształtu (ang. *shape index*) dla punktów i poszukiwanie ekstremów. Zaletą jest determinizm wyboru punktów. Ponieważ liczba wybranych punktów jest też znacznie mniejsza niż w podejściu nr 1, to koszt

obliczeniowy dopasowywania jest też znacznie niższy. Dodatkowo opisywana metoda cechuje się wszystkimi pozytywnymi właściwościami podejścia nr 1.

3) Generacja deskryptorów dla punktów wybranych losowo z analizowanej chmury. Preferuje się rozkład równomierny na całej powierzchni obiektu, choć nie jest to krytyczne dla stosowania [3]. Podejście to zastosowane zostało w związku z *shape contexts* [3–5]. Wykorzystywana jest tu kluczowa właściwość deskryptorów punktów: punkty położone blisko siebie cechuje podobna charakterystyka [2]. Najistotniejszą zaletą metody jest bardzo niski koszt obliczeniowy generacji. Dodatkowo możliwy jest zewnętrzny wybór (np. podanie jako parametr) liczby punktów, dla których nastąpi generacja. Pozwala to wpływać m.in. na koszt dopasowywania. Im jest ich mniej, tym czas dopasowywania będzie krótszy. Wadą jest brak determinizmu: zbiór wygenerowanych deskryptorów w kolejnych próbach będzie nieznacznie się różnił. Pogarsza się jakość dopasowywania: koszt przyporządkowania 1–1 dwóch zbiorów deskryptorów wygenerowanych na tym samym obrazie wejściowym jest niezerowy. Pojawia się problem względnego przesunięcia punktów, np. można sobie wyobrazić następującą sytuację:

- generowane są deskryptory we wzorcu i obiekcie,
- następuje dopasowanie minimalizujące sumaryczny koszt,
- do punktu na wierzchołku nosa we wzorcu dopasowany zostaje punkt położony na grzbiecie nosa, gdyż był on najbliższy wierzchołkowi nosa z punktów wybranych do generacji deskryptorów w obiekcie.

Algorytm wykonany został całkowicie poprawnie, jednak intuicyjnie uzyskane rozwiązanie nie wydaje się do końca satysfakcjonujące.

Bibliografia

1. Cristina Conde, Angel Serrano, Licesio J. Rodríguez-Aragón, Enrique Cabello (ed.): *3D Facial Normalization with Spin Images and Influence of Range Data Calculation over Face Verification.* Universidad Rey Juan Carlos (URJC), Escuela Superior de Ciencias Experimentales y Tecnología (ESCET), Face Recognition and Artificial Vision Group (FRAV), IEEE, 2005. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05).
2. Chin-Seng Chua, Feng Han, Yeong-Khing Ho (ed.): *3D Human Face Recognition Using Point Signature.* School of Electrical and Electronic Engineering, Nanyang Technological University, IEEE, 2000. 4th IEEE International Conference on Automatic Face and Gesture Recognition (FG'00).
3. Serge Belongie, Jitendra Malik, Jan Puzicha: *Shape matching and object recognition using shape contexts.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(24), April 2002.
4. Serge Belongie, Jitendra Malik (ed.): *Matching with Shape Contexts.* IEEE, 2000. IEEE Workshop on Contentbased Access of Image and Video Libraries (CBAIVL-2000).
5. Marian Yusuf, Tarique Haider (ed.): *Recognition of Handwritten Urdu Digits using Shape Context.* Department of Computer Science, KASBIT, Karachi, IEEE, 2004. Multitopic Conference, 2004. Proceedings of INMIC 2004. 8th International.
6. Yang Li, William A.P. Smith, Edwin R. Hancock (ed.): *Face Recognition using Patch-based Spin Images.* Department of Computer Science, University of York, IEEE, 2006. 18th International Conference on Pattern Recognition (ICPR'06).
7. Andrew E. Johnson, Martial Hebert: *Using spin images for efficient object recognition in cluttered 3D scenes.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(5), May 1999.
8. Salvadro Ruiz-Correa, Linda G. Shapiro, Marina Melia (ed.): *A New Signature-Based Method for Efficient 3-D Object Recognition.* Department of Electrical Engineering, Department of Statistics, University of Washington, IEEE, 2001.
9. Hui Chen, Bir Bhanu: *3D free-form object recognition in range images using local surface patches.* Pattern Recognition Letters, 28:1252–1262, 2007.
10. Zhnohui Wu, Yueming Wang, Gnng Pan (ed.): *3D Face Recognition Using Local Shape Map.* Department of Computer Science and Engineering Zhejiang University, Hangzhou, IEEE, 2004. 2004 International Conference on Image Processing (ICIP).
11. Chin Seng Chua, Ray Jarvis. *Point signatures: A new representation for 3D object recognition.* International Journal of Computer Vision, 25(1):63–85, 1997.
12. Hui Chen, Bir Bhanu: *Human ear recognition in 3D.* IEEE Transactions On Pattern Analysis And Machine Intelligence, 29(4), April 2007.

Automatic nose measurement system, part 4

Abstract: The purpose of designed system is to analyze and recognize three-dimensional face images. Using known techniques, algorithms and tools I am aiming to retrieve nose parameters directly from 3D scans. Current part is devoted to describe different aspects of point descriptors usage. Wide range of known approaches is explained. Several methods of neighborhood analysis are considered. Distance measure of two descriptors is introduced. Possible methods of quantization are described. At the end, application of Hungarian algorithm for descriptors matching is shown.

Keywords: anthropometry, shape recognition, descriptor, face analysis, spin images, 3D scans, nose measurements

Tomasz Kuśmierczyk

Ukończył studia pierwszego stopnia na Wydziale Elektroniki i Technik Informacyjnych PW na kierunku Informatyka. Kontynuuje studia na poziomie magisterskim. Jednocześnie studiuje na Wydziale Fizyki UW na kierunku Fizyka. Zainteresowania naukowe: sztuczna inteligencja i uczenie maszynowe w zastosowaniu do analizy i rozpoznawania obrazów.

e-mail: tk290810@okwf.fuw.edu.pl