

2. Przegląd wybranych algorytmów śledzenia obiektów

Intuicyjnie rzecz ujmując, śledzenie wybranego obiektu polega na określaniu jego pozycji w kolejnych ramach obrazu. W literaturze przedmiotu można znaleźć wiele różnych algorytmów służących do rozwiązania tego zadania, jednak każda z tych metod posiada pewne dodatkowe założenia w stosunku do śledzonego obiektu. Na początku tego rozdziału zostaną zaprezentowane podstawowe cechy opisujące obiekt, a następnie pokazane będą trzy wybrane algorytmy śledzenia: KLT, *mean shift* oraz filtr cząsteczkowy.

2.1. Obiekt

2.1.1. Reprezentacja

Pierwszym pytaniem, jakie projektant systemu śledzenia musi sobie zadać, to jaki będzie najlepszy sposób reprezentacji obiektu. Najpopularniejsze z nich to [1] (rysunek 2.1):

- Proste kształty geometryczne (elipsa, prostokąt),
- Punkt,
- Kontur,
- Sylwetka,
- Model łączony, złożony z kilku różnych połączonych ze sobą reprezentacji.

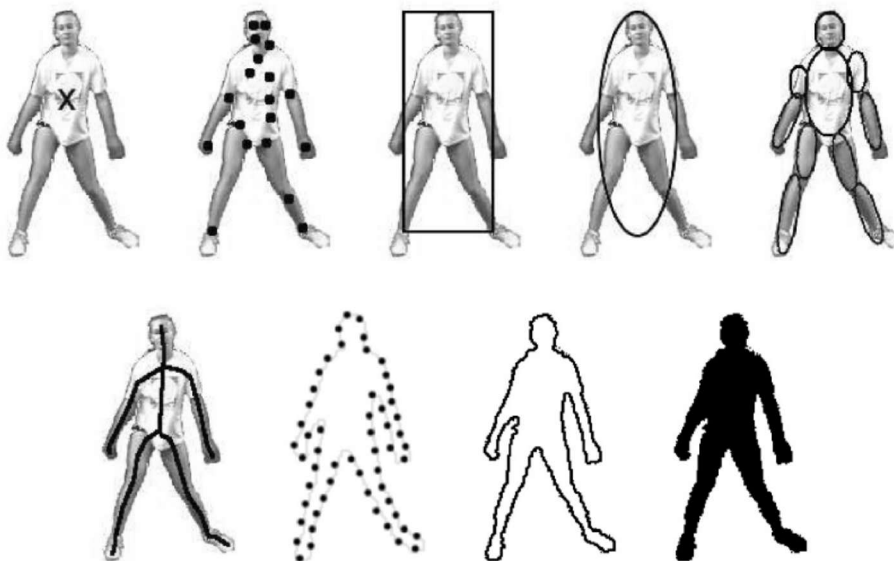
Najbardziej ogólną metodą jest śledzenie za pomocą prostych kształtów geometrycznych. Pomimo prostej reprezentacji, jest to zwykle bardzo dobre przybliżenie naszego celu. Dzięki niewielkiemu skomplikowaniu, algorytmy wykorzystujące tę reprezentację są stosunkowo szybkie.

Pozostałe możliwości pozwalają za to lepiej wykorzystywać pewne cechy charakterystyczne, jak dokładny kształt albo wielkość.

2.1.2. Kryterium śledzenia

Wybór kryterium śledzenia (ang. *tracking feature*) sprowadza się do odszukania pewnej cechy, dzięki której będzie możliwe odróżnienie obiektu od tła. Tak samo jak przy wyborze reprezentacji, odpowiedni dobór cechy pozwala skuteczniej wykorzystać wiedzę o obiekcie, albo przeciwnie, nie ograniczać się tylko do konkretnej gamy obiektów. Najczęściej spotyka się następujące kryteria: [1]

- Kolor,



Rys. 2.1. Różne sposoby reprezentacji obiektu (źródło [1])

- Krawędzie,
- Przepływ optyczny – pole opisujące przemieszczenia poszczególnych pikseli w obrazie,
- Tekstura – zbiór bardziej skomplikowanych cech charakterystycznych pewnego obiektu.

W tym przypadku, najbardziej ogólnym wyborem jest kolor obiektu, ponieważ bardzo rzadko jest on zmienny. Pewnym utrudnieniem tutaj mogą być różnice w „postrzeganiu” koloru przez kamerę, związane na przykład ze zmianą oświetlenia. Z tego powodu często stosuje się tutaj inne przestrzenie barw niż RGB, na przykład YCbCr ([2]) albo HSV (ang. *Hue Saturation Value*) ([3]), a matematyczną reprezentacją będzie odpowiedni histogram.

2.2. Opis wybranych algorytmów

2.2.1. KLT

KLT to jeden ze starszych algorytmów śledzenia obiektów. Jego podstawowa wersja wykorzystuje reprezentację punktową, a kryterium śledzenia jest przepływ optyczny. Całkowicie ignorowany jest kolor obiektu, ponieważ algorytm operuje na obrazie w skali szarości. Nazwa pochodzi od nazwisk jego twórców: Takeo Kanade, Bruce’a Lucasa oraz Carlo Tomasi. Jego pierwsza wersja została zaprezentowana w pracy [4] z roku 1981, a pewne ulepszenia zostały opisane w artykule [5] z roku 1991.

Skrócony opis algorytmu

Podstawowym zadaniem jest znalezienie takich przesunięć ξ oraz η , aby spełniona była następująca równość (przepływ optyczny):

$$I(x, y, t + \tau) = I(x - \xi, y - \eta, t) \quad (2.1)$$

gdzie:

x – położenie punktu na osi x

y – położenie punktu na osi y

t – czas

τ – odstęp czasu

ξ – przesunięcie obiektu wzdłuż osi x

η – przesunięcie obiektu wzdłuż osi y

$I(x, y, t)$ – pewna funkcja (reprezentująca natężenie światła w punkcie (x, y) w czasie t)

W dalszej części przyjmijmy $\mathbf{d} = (\xi, \eta)$, $\mathbf{x} = (x, y)$, $J(\mathbf{x}) = I(x, y, t + \tau)$.

Oczywistym jest, że w rzeczywistych przypadkach równość ta nie zostanie nigdy spełniona – natężenie światła będzie się zmieniało przykładowo ze względu na zróżnicowany poziom oświetlenia. Z tego powodu należy przeprowadzić poszukiwania takiego \mathbf{d} w pewnym obszarze \mathcal{W} , aby zminimalizować błąd dany wzorem:

$$\epsilon = \int_{\mathcal{W}} [I(\mathbf{x} - \mathbf{d}) - J(\mathbf{x})]^2 d\mathbf{x} \quad (2.2)$$

Ze względu na fakt, że dwie kolejne ramki obrazu niewiele się różnią od siebie, można rozwinąć funkcję $I(\mathbf{x} - \mathbf{d})$ za pomocą szeregu Taylora:

$$I(\mathbf{x} - \mathbf{d}) = I(\mathbf{x}) - \mathbf{g} \cdot \mathbf{d}$$

gdzie:

\mathbf{g} – gradient funkcji I w punkcie \mathbf{x}

Po pewnych przekształceniach można otrzymać, że wartość \mathbf{d} minimalizująca równanie (2.2) jest rozwiązaniem następującego równania liniowego:

$$G\mathbf{d} = \mathbf{e} \quad (2.3)$$

gdzie:

G – macierz dana wzorem: $G = \int_{\mathcal{W}} \mathbf{g}\mathbf{g}^T d\mathbf{x}$

\mathbf{e} – wektor dany wzorem: $\mathbf{e} = \int_{\mathcal{W}} (I - J) \mathbf{g} d\mathbf{x}$

Wyliczony w ten sposób punkt $\mathbf{x} + \mathbf{d}$ jest szukaną przez nas pozycją obiektu w kolejnej ramce obrazu. Przedstawione powyżej wyprowadzenie sprawdza się dla ciągłej w przestrzeni funkcji natężenia światła. Jednak w przypadku komputerowego śledzenia obiektów nie jesteśmy w stanie takiej funkcji otrzymać. Z tego powodu wykorzystuje się dyskretną wersję tego równania, w której całki są zamieniane na odpowiadające im sumy.

Pełny algorytm prezentuje się więc następująco:

1. Konwersja obrazu do skali szarości (na przykład obliczenie składowej Y z przestrzeni YCbCr).
2. Określenie wielkości okna \mathcal{W} , zazwyczaj kwadrat o długości boku równym kilka pikseli.
3. Obliczenie gradientu \mathbf{g} w każdym punkcie okna.
4. Wyznaczenie macierzy G oraz wektora \mathbf{e} .
5. Rozwiązanie równania liniowego (2.2), aby otrzymać przesunięcie obiektu \mathbf{d} .
6. Wyznaczenie nowej pozycji obiektu równej $\mathbf{x} + \mathbf{d}$.

2.2.2. Mean shift

Mean shift jest to nieparametryczny algorytm służący do znajdowania maksimum funkcji gęstości prawdopodobieństwa. Ze względu na swoją wydajność, znalazł swoje miejsce jako jeden z najpopularniejszych algorytmów w wizyjnym śledzeniu obiektów. Pierwszy opis jego użycia w tym celu znajduje się w pracy [6]. Wykorzystano tam reprezentację za pomocą elipsy, a śledzoną cechą był kolor obiektu (dokładniej jego histogram).

Skrócony opis algorytmu

Pierwszym punktem algorytmu jest uzyskanie matematycznego opisu śledzonego obiektu, co należy zrobić w pierwszej ramce obrazu. Wykorzystuje się do tego model celu dany wzorem:

$$\hat{\mathbf{q}} = \{\hat{q}_u\}_{u=1\dots m} \quad (2.4)$$

gdzie:

m – rozmiar histogramu

Model ten musi spełniać założenia funkcji gęstości prawdopodobieństwa, więc:

$$\sum_{u=1}^m \hat{q}_u = 1 \quad (2.5)$$

Niech $\{\mathbf{x}_i^*\}_{i=1\dots n}$ będą lokalizacjami wszystkich pikseli znajdującymi się wewnątrz śledzonego obiektu, wcześniej znormalizowanymi do koła jednostkowego o środku w punkcie $(0, 0)$. Normalizacja ta jest niezbędna, aby uniezależnić wszystkie dalsze obliczenia od kształtu oraz pozycji obiektu. Wtedy poszczególne składowe modelu można obliczyć za pomocą wzoru:

$$\hat{q}_u = C \sum_{i=1}^n k(\|\mathbf{x}_i^*\|^2) \delta[b(\mathbf{x}_i^*) - u] \quad (2.6)$$

gdzie:

δ – delta Kroneckera

$k(x)$ – profil wybranego jądra (funkcja, która daje większą wagę punktom znajdującym się bliżej środka obiektu)

$b(x)$ – funkcja zwracająca, w której części histogramu znajduje się punkt x

C – stała normalizująca tak, aby prawdziwa była zależność (2.5).

Model celu jest więc w rzeczywistości trochę zmodyfikowanym histogramem, do którego z większą wagą wliczane są punkty znajdujące się blisko środka obiektu.

W kolejnych ramkach obrazu należy wyznaczać kandydatów do celu danych wzorem:

$$\hat{\mathbf{p}}(\mathbf{y}) = \{\hat{p}_u(\mathbf{y})\}_{u=1\dots m} \quad (2.7)$$

gdzie:

\mathbf{y} – znormalizowana pozycja kandydata

Podobnie, jak w przypadku modelu, suma wszystkich składowych musi się równać 1:

$$\sum_{u=1}^m \hat{p}_u = 1 \quad (2.8)$$

Poszczególne części składowe kandydata do celu dane są wzorem:

$$\hat{p}_u(\mathbf{u}) = C_h \sum_{i=1}^{n_h} k\left(\left\|\frac{\mathbf{y} - \mathbf{x}_i}{h}\right\|^2\right) \delta[b(\mathbf{x}_i) - u] \quad (2.9)$$

gdzie:

h – skala obiektu

C_h – stała normalizująca

Funkcja jądra może być praktycznie dowolna, jednak ze względów praktycznych najczęściej wykorzystuje się jądro z profilem Epanechnikova dane wzorem:

$$k(x) = \begin{cases} 1 - x & \text{jeśli } x \leq 1 \\ 0 & \text{w przeciwnych wypadkach} \end{cases} \quad (2.10)$$

Jakość kandydata określana jest za pomocą jego odległości od modelu:

$$d(\mathbf{y}) = \sqrt{1 - \rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}]} \quad (2.11)$$

gdzie:

$$\hat{\rho}(\mathbf{y}) = \rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}] \quad (2.12)$$

to współczynnik Bhattacharyya pomiędzy \mathbf{p} oraz \mathbf{q} .

Obiekt w kolejnej ramce obrazu znajduje się więc w takim punkcie \mathbf{y} , gdzie współczynnik Bhattacharyya jest największy. W tym celu wykorzystuje się właściwy algorytm optymalizacji *mean shift*.

Bez wchodzenia w szczegóły wyprowadzeń, metoda ta przedstawia się następująco:

1. Ustalenie punktu początkowego $\hat{\mathbf{y}}_0 = \hat{\mathbf{x}}$
2. Obliczenie wag poszczególnych części histogramu zgodnie ze wzorem:

$$w_i = \sum_{u=1}^m \sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\mathbf{y}}_0)}} \delta[b(\mathbf{x}_i) - u] \quad (2.13)$$

3. Odszukanie kolejnej lokalizacji dla kandydata zgodnie ze wzorem:

$$\hat{\mathbf{y}}_1 = \frac{\sum_{i=1}^{n_h} \mathbf{x}_i w_i}{\sum_{i=1}^{n_h} w_i} \quad (2.14)$$

4. Zakończenie poszukiwań, jeśli odległość pomiędzy punktami $\hat{\mathbf{y}}_1$ oraz $\hat{\mathbf{y}}_0$ jest mniejsza niż ustalony wcześniej próg ϵ . Wartość ϵ ustala się zazwyczaj tak, by punkty znajdowały się wewnątrz tego samego piksela.
5. W przeciwnym wypadku podstawienie $\hat{\mathbf{y}}_0 = \hat{\mathbf{y}}_1$ oraz powrót do kroku 2.

Aby dodatkowo uzyskiwać informację o zmieniającym się rozmiarze obiektu, przeprowadza się powyższy algorytm trzykrotnie dla różnych wartości współczynnika h . Niech h_{prev} będzie skalą w poprzedniej ramce. Wtedy zazwyczaj przeszukuje się następujące skale: $0.9h_{prev}$, h_{prev} oraz $1.1h_{prev}$. Niech h_{opt} będzie wynikiem, dla którego osiągnięto najwyższy współczynnik podobieństwa modelu do kandydata. Wtedy za skalę w obecnej ramce uznaje się wartość: $h_{new} = 0.1h_{opt} + 0.9h_{prev}$. Tak przeprowadzane obliczenia rozmiaru pozwalają na wykorzystywanie mniejszych zasobów obliczeniowych.

2.2.3. Filtry cząsteczkowe

Filtry cząsteczkowe (ang. *particle filters*), inaczej sekwencyjne metody Monte Carlo, reprezentują kolejną grupę ogólniejszych algorytmów, które znalazły swoje zastosowanie w dziedzinie śledzenia obiektów. W przeciwieństwie do poprzednio opisywanych, są one oparte na prawdopodobieństwie znajdowania się obiektu w danym punkcie. Dodatkowo, wykorzystują one losowe rozmieszczanie swoich cząsteczek zgodnie z pewnym rozkładem prawdopodobieństwa, więc są niedeterministyczne. Wykorzystywany tutaj algorytm ma swoje źródło w pracy [7]. Obiekt jest w nim reprezentowany za pomocą pewnego dowolnego, prostego kształtu (elipsa bądź prostokąt), a śledzoną cechą jest kolor.

Skrócony opis algorytmu

Założmy, że śledzony obiekt porusza się zgodnie z pewnym równaniem stanu i jest obserwowany za pomocą niezależnego procesu. Przyjmijmy, że \mathbf{x}_t reprezentuje ukryty, prawdziwy stan obiektu w chwili t , a \mathbf{y}_t to ten obserwowany w chwili t . Algorytm opiera się o wyprowadzoną z twierdzenia Bayesa zależność:

$$p(\mathbf{x}_{t+1}|\mathbf{y}_{0:t+1}) \propto p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}) \int_{\mathbf{x}_t} p(\mathbf{x}_{t+1}|\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{y}_{0:t}) d\mathbf{x}_t \quad (2.15)$$

gdzie: zapis $\mathbf{x}_{0:t}$ oznacza wektor $(\mathbf{x}_0, \dots, \mathbf{x}_t)$. Lewa strona tego równania to rozkład prawdopodobieństwa, że szukany obiekt znajduje się w miejscu \mathbf{x}_{t+1} wiedząc, że pomiary dotychczas wskazywały $\mathbf{y}_{0:t+1}$. Wartość oczekiwana tego rozkładu reprezentuje najbardziej prawdopodobną pozycję, którą przyjmuje się jako wynik w zadaniu śledzenia.

Podstawą do rozwiązania powyższego równania w niniejszym algorytmie jest metoda Monte Carlo, w której skomplikowane problemy przybliżane są za pomocą skończonego zbioru M losowo rozmieszczonych cząsteczek $\{\mathbf{x}_t^m\}_{m=1 \dots M}$. W tym przypadku, przybliżane jest prawdopodobieństwo $p(\mathbf{x}_{t+1}|\mathbf{y}_{0:t+1})$, a cząsteczki reprezentują możliwe lokalizacje śledzonego obiektu. Aby możliwe było skorzystanie ze wzoru (2.15) należy przekształcić elementy z prawej strony równania na operacje na cząsteczkach.

Rozkład $p(\mathbf{x}_t|\mathbf{y}_{0:t})$ jest wynikiem działania algorytmu w poprzedniej ramce obrazu, więc jego przybliżenie w każdej ramce obrazu poza pierwszą jest dostępne. W pierwszej ramce należy przyjąć z góry pewien odpowiednio dobrany rozkład – zazwyczaj jednostajny albo normalny. W każdej iteracji algorytmu występuje losowanie pozycji cząsteczek zgodnie z tym rozkładem – w pierwszej rozmieszcza się je w całej przestrzeni rozwiązań, a w kolejnych wybierane są z już istniejących.

Prawdopodobieństwo $p(\mathbf{x}_{t+1}|\mathbf{x}_t)$ reprezentowane jest za pomocą ruchu cząsteczek zgodnie z równaniem stanu obiektu danym wzorem:

$$\mathbf{x}_{t+1} = A\mathbf{x}_t + B\mathbf{x}_{t-1} + v_t \quad (2.16)$$

gdzie: $v_t \sim \mathcal{N}(0, C)$ jest to pewna losowa wartość wygenerowana z rozkładu normalnego o wartości oczekiwanej równej 0. Jako stan obiektu przyjmuje się zazwyczaj jego lokalizację, czasami wraz z rozmiarem. Wartości macierzy A , B ustalone są tak, aby przedstawiały model ruchu o stałej prędkości, a więc $A = 2I$ oraz $B = -I$, a C można traktować jako parametr algorytmu.

Kolejną czynnością jest przypisanie każdej cząsteczce wagi, która reprezentuje w ten sposób prawdopodobieństwo $p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1})$. Jest ono zazwyczaj określane za pomocą wzoru:

$$w_i = ke^{-\Delta D^2} \quad (2.17)$$

gdzie:

k – stała normalizująca sumę wag do 1,

Λ – pewien ustalony parametr,

D – dystans.

Najczęściej stosowanym w tym algorytmie dystansem jest podobieństwo histogramów w cząsteczkach z bazowym, czyli tym obliczonym w pierwszej ramce obrazu. W celu jego obliczenia można przykładowo zastosować wykorzystywany w algorytmie *mean shift* współczynnik Bhattacharrya – wzór (2.12).

Obliczone wagi wraz z lokalizacją cząsteczek przybliżają szukany we wzorze (2.15) rozkład prawdopodobieństwa. W ostatnim kroku algorytmu wagi wykorzystywane są więc do obliczenia najbardziej prawdopodobnej pozycji śledzonego obiektu zgodnie ze wzorem:

$$\mathbf{x}_{t+1} = \sum_{i=1}^M w_{t+1}^i \mathbf{x}_{t+1}^i \quad (2.18)$$

oraz jako prawdopodobieństwo wylosowania cząsteczki podczas etapu wyboru.

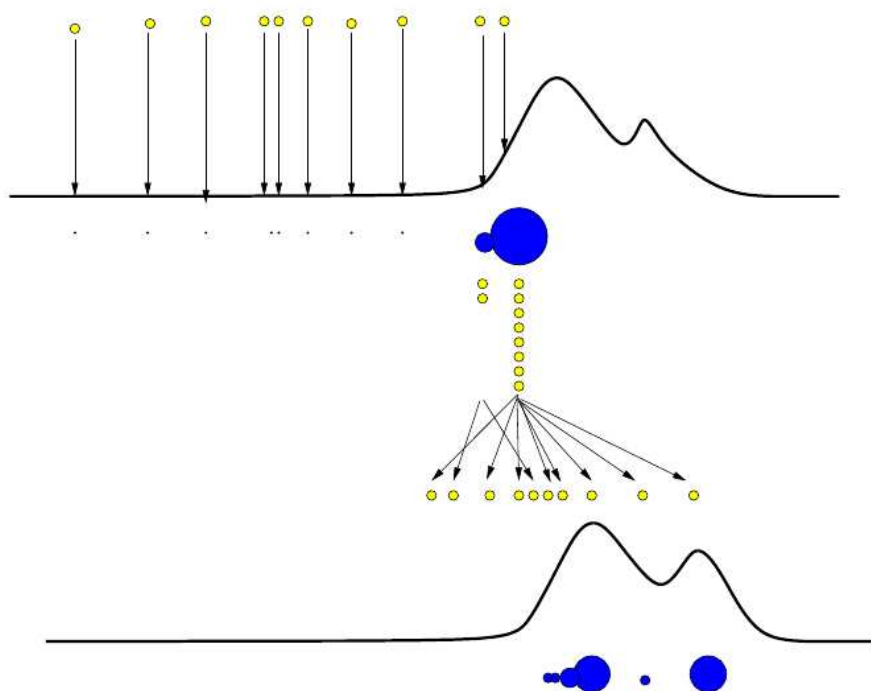
Podsumowując, w pierwszej ramce obrazu do wykonania są dwa zadania:

1. Obliczenie histogramu śledzonego obiektu.
2. Rozmieszczenie zgodnie z przewidywanym rozkładem prawdopodobieństwa cząsteczek w pobliżu obiektu.
Najczęściej stosuje się rozkład jednostajny albo normalny.

W kolejnych ramkach obrazu algorytm prezentuje się następująco:

1. Przewidywanie: przesunięcie cząsteczek zgodnie z równaniem stanu – wzór (2.16).
2. Obliczenie histogramu w lokalizacji każdej cząsteczki ze zbioru.
3. Ważenie: obliczenie wagi każdej z cząsteczek zgodnie ze wzorem (2.17).
4. Wyjście: obliczenie najbardziej prawdopodobnego stanu śledzonego obiektu za pomocą wzoru (2.18)
5. Wybór: wybranie M cząstek zgodnie z prawdopodobieństwem określonym poprzez obliczone wagi.

Graficznie etapy te zaprezentowano na rysunku 2.2, gdzie za pomocą filtru cząsteczkowego przybliżana zmieniająca się jest pewna funkcja ciągła. Pierwszy zaprezentowany etap to ważenie, po którym nastąpił wybór spośród cząsteczek o największej wadze. W dolnej części zaprezentowano, w jaki sposób może działać etap przewidywania.



Rys. 2.2. Działanie filtru cząsteczkowego (źródło [8])