Ćwiczenie 3: Detekcja ruchu

Adrian Jałoszewski

23 października 2017

1 Operacja SAD

SAD – sum of absolute differences – suma wartości bezwzględnych z różnicy pikseli dwóch kolejnych ramek sekwencji wideo.

$$d_\tau(I_t, I_{t-1}) = \sum_{x,y} |I_t(x,y) - I_{t-1}(x,y)|$$

W przypadku ruchu zmieniają się piksele obrazu, dlatego też zmienia się suma modułów ich różnic. Algorytm ten może być uogólniony z modułu na normę – można wtedy traktować poszczególne piksele jako wektory w przestrzeni trójwymiarowej. Aby uniezależnić ruch od oświetlenia można zastosować np. transformację RGB na YCbCr i potraktować składowe Cb oraz Cr jako wektor dwuwymiarowy.

2 Motion Threshold dla SAD oraz parametry bloków Add i Abs

Motion Threshold dla operacji SAD powinien wynosić 320 000 - jest to wartość powyżej której na obrazie występował ruch.

Na wejściu bloku Add znajdują się dwie liczby całkowite 8-bit bez znaku, przez co mają zakres od 0 do 255. Po odjęciu skrajnych wartości od siebie możemy mieć wynik w zakresie od -255 do 255, co można zapisać na liczbie siedmiobitowej ze znakiem, dlatego też powinna być tu ustawiona liczba całkowita ze znakiem o 16 bitach. Modół z niej ma już zasięg tylko od 0 do 255, więc blok abs powinien przyjmować liczby całkowite 16-bit ze znakiem, a na wyjściu mieć liczby całkowite 8-bit bez znaku.

3 Uniezależnienie wyników od zakłóceń i zwiększenie skuteczności detekcji

Aby uniezależnić wynik od zakłóceń można wstępnie przeprowadzić na obrazie filtrację medianową o stosunkowo małym rozmiarze (3x3 – 5x5), pozwoli to na usunięcie szumów występujących na obrazie.

Równie dobrym pomysłem jest zastosowanie ważonej średniej ruchomej pomiędzy pikselami ze współczynnikami wykładniczymi. Ze względu na szybki zanik wartości z początku nie wymagałoby to ich odejmowania. Jednak złe dobranie parametrów może tu poskutkować opóźnieniem w pojawieniu się ruchu – ruch będzie dopiero obserwowalny po kilku klatkach.

4 Parametry τ oraz Motion Threshold

Algorytm MHI działa dobrze dla następujących parametrów:

– $\tau = 256$ oraz Motion Threshold = 28

– $\tau = 128$ oraz Motion Threshold = 40

Gdzie parametr Motion Threshold jest minimalną różnicą przy której wykrywany jest ruch – zbyt małe różnice nie są traktowane jako ruch. Parametr τ jest informacją o zanikaniu ruchu – przez tyle iteracji w przypadku niewykrycia ruchu dla danej komórki jest on wygaszany (odejmowana jest jedynka).

Jako uwaga boczna – jest to dosyć sprytne podejście do algorytmu, które wymaga przemyślenia. Zdecydowanie lepiej by było dla tej implementacji MHI jakby zamiast mnożenia i dodawania były tam zwykłe instrukcje warunkowe.

5 Inny algorytm – Lucas-Kanade

Algorytmy przepływu optycznego (optical flow) – jest to grupa algorytmów służących do wykrywania względnego ruchu pomiędzy obserwatorem i sceną. Są one wykorzystywane do wykrywania ruchu w systemach wizyjnych.

Metody te mają na celu wyznaczenie rucchu pomiędzy dwiema ramkami obrazu w chwilach czasowych t oraz $t + \Delta t$ dla każdego z wokseli obrazu trójwymiarowego. Poprzez zastosowanie przekroji może to być wykorzystane do wyznaczenia ruchu na płaszczyźnie.

Przypadek dwówymiarowy jest określony przez równanie:

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t)$$

Oznacza to, że jasność pewnego piksela w czasie w pewnym punkcie po jego przesunięciu pozostaje stała (piksel zostal przesunięty w czasie Δt o wektor $[\Delta x, \Delta y]$, nie tracąc przy tym na jasności. Rozwijając te równanie w szereg taylora dla małych przesunięć w czasie mamy:

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + R(x, y, t)$$

Gdzie R jest dla małych różnic w czasie pomijalnie małe. Wynika z tego, że:

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0$$

Dzieląc przez Δt

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t} = 0$$

Dla bardzo małych różnic czasu można zastosować przybliżenie:

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} = 0$$

Należy dla każdego piksela wyznaczyć V_x oraz V_y. Ze względu na to, że równanie te ma dwie niewiadome stosuje się dodatkowe ograniczenia, które mają na celu ujednoznacznić wynik. Np. Metoda Lucasa-Kanade, która korzysta z macierzy pseudoodwrotnej.