PRZETWARZANIE OBRAZÓW CYFROWYCH 2

ĆWICZENIE 1

ALGORYTMY PRZETWARZANIA WSTĘPNEGO - FILTRY NIELINIOWE

FILTRACJA OBRAZÓW Z POZIOMAMI SZAROŚCI

W poprzednich ćwiczeniach omawiane były metody punktowego przetwarzania obrazów takie jak operacje na histogramie umożliwiały one poprawę kontrastu, uwypuklenie pewnych detali obrazu. W kolejnym ćwiczeniu wprowadzono podstawowe filtry kontekstowe, których wyjściem była liniowa kombinacją otoczenia przetwarzanego piksela, wprowadzono filtry konwolucyjne umożliwiające wyostrzanie i rozmywanie obrazów. Z wykorzystaniem filtrów liniowych możliwe efektywne było usuwanie zakłóceń z obrazu, szczególnie gdy miały one charakter addytywny, niestety wiązało się to ze znaczną degradacją przetwarzanego obrazu (rozmyciem). Powstały więc liczne metody nieliniowe które miały ograniczyć nadmierne rozmywanie obrazu oraz poprawić efektywność redukcji zakłóceń impulsowych.

Powstało wiele algorytmów będących modyfikacjami filtrów liniowych np. **uśrednianie z walidacją danych**, polegającą na uśrednianiu tylko punktów spełniających określone kryterium. Kolejnym możliwym rozwiązaniem poprawiającym filtrację liniową jest **uśrednianie wg odwrotnego gradientu**.

Najczęściej obecnie używanymi metodami nieliniowymi są filtry wykorzystujące statystyki porządkowe np. filtr medianowy, minimalny, maksymalny.

FILTR MEDIANOWY, MINIMALNY I MAKSYMALNY

Filtr medianowy jako wyjście filtra wybiera medianę z otoczenia przetwarzanego punktu, zakładając relację 8-sąsiedztwa dla maski o rozmiarze 3x3 otrzymamy:

$$\begin{array}{|ccc|} \hline x_1 & x_2 & x_3 \\ x_4 & x_0 & x_5 \\ x_6 & x_7 & x_8 \\ \hline \end{array}$$

Gdzie x_0 jest przetwarzanym punktem.

W celu wyznaczenia wyjścia filtra medianowego musimy uporządkować wartości pikseli:

$$x_{(0)} \le x_{(1)} \le x_{(2)} \le x_{(3)} \le x_{(4)} \le x_{(5)} \le x_{(6)} \le x_{(7)} \le x_{(8)}.$$

W przypadku filtru medianowego oryginalny piksel x_0 zostaje zastąpiony przez wartość $x_{(4)}$. Dla filtrów maksymalnego i minimalnego wyjście filtra definiowane jest odpowiednio jako maksimum i minimum z maski czyli $x_{(8)}$ oraz $x_{(0)}$.

Działanie filtru medianowego w porównaniu do zwykłego uśredniania przedstawiono na poniższej ilustracji

1. Obraz oryginalny zakłócony 10% szumem impulsowym

2. Wynik uśredniania w oknie 3x3

3. Filtr medianowy dla okna 3x3

FILTR LUM

Filtr medianowy efektywnie usuwa nawet silne zakłócenia impulsowe, niestety dla obrazów słabo zakłóconych „niepotrzebnie" zmienia także wiele „właściwych" punktów powodując tym samym niepotrzebne rozmycie i utratę detali.

Aby zapobiec nadmiernej degradacji obrazów zaproponowano modyfikację filtru medianowego, w której piksele bliskie medianie uznajemy za niezakłócone i nie zmieniamy ich wartości. W tym celu, podobnie jak dla filtru medianowego musimy dokonać sortowania punktów z maski, dla ułatwienia ograniczymy się do rozmiaru maski 3x3.

$$\boxed{\begin{array}{ccc} x_1 & x_2 & x_3 \\ x_4 & x_0 & x_5 \\ x_6 & x_7 & x_8 \end{array}}$$

$$x_{(0)} \le x_{(1)} \le x_{(2)} \le x_{(3)} \le x_{(4)} \le x_{(5)} \le x_{(6)} \le x_{(7)} \le x_{(8)}.$$

Załóżmy, że przez k oznaczymy rozmiar otoczenia mediany uznawany za niezakłócony, uzyskamy w ten sposób dwie wartości progowe $x_{(4)-k} = x_L$ (Lower) oraz $x_{(4)+k} = x_U$ (Upper), zakładając, że $k = 2$ możemy zapisać:

$$x_{(0)} \le x_{(1)} \le x_L \le x_{(3)} \le x_{MED} \le x_{(5)} \le x_U \le x_{(7)} \le x_{(8)}.$$

Wyjście filtra LUM definiujemy jako:

$$x_{LUM} = med\left(x_L, x_U, x_0\right),$$

gdzie x_0 oznacza punkt centralny maski.

FILTRACJA OBRAZÓW BARWNYCH

W obrazach barwnych pojedynczy piksel jest najczęściej zapisywany za pomocą trzech wartości R, G oraz B. Cały obraz jest więc macierzą wektorów RGB, bezpośrednie zastosowanie metod filtracji opracowanych dla obrazów z poziomami szarości nie zawsze jest możliwe.

Najprostszym podejściem jest potraktowanie obrazu barwnego jako trzy osobne obrazy z poziomami szarości (odpowiednio dla kanałów R, G oraz B), a następnie zastosowanie metody opracowanej dla obrazów jednokanałowych. Kanały R, G, B nie są jednak niezależne – w obrazach rzeczywistych można zaobserwować wysoką korelację między poszczególnymi składowymi barwnymi. Zaproponowano więc różne techniki uwzględniające wektorowy charakter przetwarzanego sygnału. W przypadku filtrów rangowych sposób uporządkowania wektorów nie jest sprawą banalną, najbardziej popularne jest zastosowanie tak zwanego zredukowanego porządkowania zastosowanego między innym w filtrze VMF (Vector Median Filter) [1].

OPIS DZIAŁANIA FILTRU VMF (VECTOR MEDIAN FILTER)

Filtr mediany wektorowej (VMF) jest jednym z podstawowych reprezentantów rodziny wektorowych operatorów statystycznych. Jest on rozwinięciem klasycznej mediany skalarnej na przestrzeń wektorową zaproponowaną w pracy [1] . W wielowymiarowych przestrzeniach wektorowych nie istnieje jednoznaczna i uniwersalna metoda definiowania porządku. Zaproponowano kilka sposobów sortowania danych wektorowych takich jak:

- Porządek brzegowy - marginal ordering (M-ordering) w którym dane porządkowane są niezależnie dla każdego wymiaru
- Porządek warunkowy - conditional ordering (C-ordering) – w którym dane wielowymiarowe zostają uporządkowane ze względu na wybraną składową.

- Porządek zredukowanego sumowania - reduced or aggregate ordering (R-ordering), w którym dane wielowymiarowe zostają zredukowane do pewnej wartości skalarnej według określonej metryki.

W wektorowym filtrze medianowym do sortowania wektorów wykorzystano porządkowanie zredukowane (R-ordering).

Załóżmy, że przez $\mathbf{F}(x)$ oznaczymy wielokanałowy obraz barwny przez W oznaczymy okno filtracji o rozmiarze k (dla maski 3×3 $k=9$). Wektory obrazu zakłóconego w oknie filtracji W oznaczymy jako $\mathbf{F}_{\mathrm{j}}$, $j=0,1,\ldots k-1$. Jeśli dystans między wektorami $\mathbf{F}_i$ oraz $\mathbf{F}_j$ oznaczymy jako $\rho\left(\mathbf{F}_i,\mathbf{F}_j\right)$ wówczas zagregowany dystans dla wektora $\mathbf{F}_i$ możemy zapisać:

$$R_i=\sum_{j=0}^{k-1}\rho\left(\mathbf{F}_{\mathrm{i}},\mathbf{F}_{\mathrm{j}}\right)$$

Możemy założyć, że odpowiedniemu posortowaniu skalarnych wartości R_i :

$$R_{(0)}\le R_{(1)}\le\ldots\le R_{(\tau)}\le\ldots\le R_{(k-1)},$$

Odpowiada takie samo uporządkowanie wektorów $\mathbf{F}_i$:

$$\mathbf{F}_{(0)},\mathbf{F}_{(1)},\ldots,\mathbf{F}_{(\tau)},\ldots,\mathbf{F}_{(n-1)}.$$

Wyjściem filtra VMF będzie więc wektor najbardziej centralnie położony, czyli $\mathbf{F}_{VM}=\mathbf{F}_{(0)}$. Można zauważyć, że spełniona jest zależność:

$$\sum_{j=0}^{k-1}\rho(\mathbf{F}_{VM},\mathbf{F}_j)<\sum_{j=0}^{k-1}\rho(\mathbf{F}_i,\mathbf{F}_j),\quad i=0,\ldots,k-1.$$

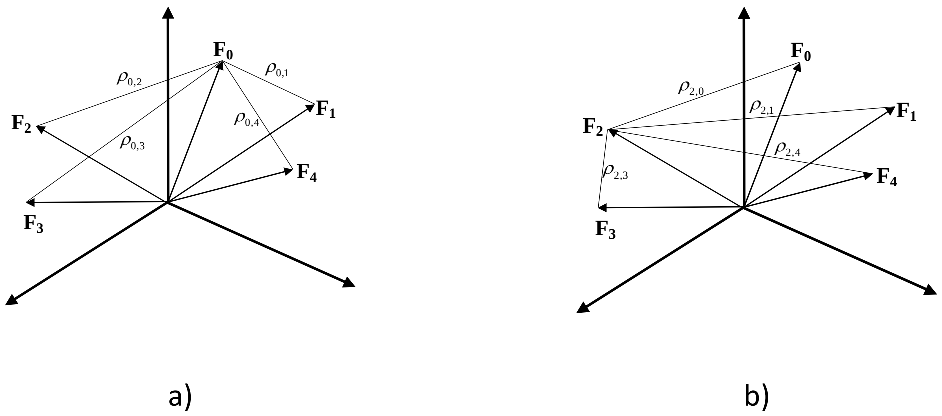
Ważnym parametrem podczas konstrukcji opisywanego filtra jest wybór metryki wykorzystywanej do liczenia odległości między wektorami $\rho\left(\mathbf{F}_i,\mathbf{F}_j\right)$ - zazwyczaj jest to metryka Minkowskiego L_p opisana zależnością:

$$\rho(\mathbf{F}_i,\mathbf{F}_j)=\left\{\sum_{k=1}^{l}(F_i^k-F_j^k)^p\right\}^{\frac{1}{p}}$$

Wybierając odpowiednią wartość p otrzymamy najczęściej używane normy L_1,L_2,L_∞ :

$$\rho(\mathbf{F}_i,\mathbf{F}_j)=\sum_{k=1}^{l}\left|(F_i^k-F_j^k)^p\right|,\ \ p=1$$
$$\rho(\mathbf{F}_i,\mathbf{F}_j)=\sqrt{\sum_{k=1}^{l}(F_i^k-F_j^k)^2},\ \ p=2$$
$$\rho(\mathbf{F}_i,\mathbf{F}_j)=\max_k\left|F_i^k-F_j^k\right|,\ \ p=\infty$$

Na poniższym rysunku przedstawiono sposób liczenia zagregowanych odległości dla dwóch wybranych wektorów przy użyciu normy Euklidesowej (L_2) i okna filtracji o rozmiarze $k=5$:



$$R_0=\rho(0,1)+\rho(0,2)+\rho(0,3)+\rho(0,4)$$
$$R_2=\rho(2,0)+\rho(2,1)+\rho(2,3)+\rho(2,4)$$

[1] J. Astola, H. Avisto, and Y. Neuovo. Vector Median Filters. In IEEE Proc., vol. 78,p. 678-689, 1990