

5. Линейни филтри за обработка на полутонови изображения. Представяне в обектното и в честотното пространство. Диференциални филтри. Нелинейни филтри.

Целта на филтрацията е да намали случайните шумове при формирането на функцията на изображението. Представянето на функцията на изображението $f(x,y)$ в паметта на компютъра по същество представлява числов масив $f(i,j)$, $i \in 1,2,\dots, N$, $j \in 1, 2, M$, където $N \times M$ определя броя на пикселите на входното поле. Хистограмата е универсална характеристика за дадено изображение. Тя показва броя на пикселите, имащи еднакъв интензитет на осветеност. Може да се използва за определяне на праг при праговите алгоритми за отстраняване на шумове. Сравнявайки последователно снемани хистограми за едно и също изображение в кратък интервал от време, могат да се търсят причините за шумовете, породени от работата на апаратурата. При наличие на шум в ниските стойности на интензитета на изображението, подходящо може да бъде прилагането на бинаризация. Когато обаче шумът се намира в по-високите интензитети, се използва филтрация.

Представяне в обектното и в честотното пространство

В обработката на изображения, филтрите се използват главно, за да подтискат високите честоти в изображението, т.е. да го изгладят, или ниските честоти, т.е. да подсилят или открият граници на обекти в изображението.

Дадено изображение може да се филтрира както в честотното, така и в обектното пространство.

Първото включва трансформирането на изображението в честотното пространство, умножаването му с честотния филтър и обратната трансформация на резултата в обектното пространство. За трансформацията се използва интегралното преобразуване на Фурие, което трансформира изменението на интензитета за изображението в област на пространствена честота. Физическият смисъл на пространствените честоти е, че колкото по-рязко се мени стойността на функцията на интензитета $f(i,j)$, толкова по-големи са те. Границите и ръбовете

Функцията на филтъра в честотното пространство се оформя по такъв начин, че да отслаби някои честоти и да подсили други. Например, обикновен нискочестотен филтър (който подминава ниските честоти и запазва високите) запазва честоти, които са по-големи от даден честотен праг и премахва (поставя 0 на) всички останали.

Преминаването от обектното в честотното пространство става посредством дискретното преобразуване на Фурие. То е важен инструмент в обработката на изображения. Използва се за разлагането на изображение на неговите синусови и косинусови компоненти. Резултатът от трансформацията представлява изображението в

честотното пространство на Фурие. Входното изображение е в обектното пространство. В честотното пространство, всяка точка представлява определена честота, която се съдържа в изображението в обектното пространство.

Формулата на дискретното преобразуване на Фурие е следната:

$$F(k,l) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(i,j) e^{-i'2\pi(\frac{ki}{N} + \frac{lj}{N})}$$
, където $i' = \sqrt{-1}$ - имагинерната единица, $f(i,j)$ - функцията на интензитета на изображението, а k и l са пространствените честоти.

На практика $e^{-i'2\pi(\frac{ki}{N} + \frac{lj}{N})}$ представлява сума от синуси и косинуси, тъй като $e^{i't} = \cos t + i' \sin t$.

Аналогичният процес в обектното пространство включва извършване на конволюция на входното изображение $f(i,j)$ с филтрираща функция $h(i,j)$. Това може да се запише като: $g(i,j) = f(i,j) * h(i,j)$. Операцията конволюция определя действие умножение над съответните стойности на функции с определена размерност (обикновено различни за отделните функции), но в едно и също пространство като резултатите от умножението се сумират за дискретните функции или интегрират за непрекъснатите.

Математическата операция конволюция в обектното пространство е идентична на умножението в честотното пространство, но резултатът от цифровата реализация варира, т.к. се налага да работим с приближение на филтриращата функция с дискретно и крайно ядро.

Дискретната конволюция може да се дефинира като „мести и умножавай“, като плъзгаме ядрото по изображението и умножаваме неговата стойност със стойността на съответния пиксел от изображението. За квадратно ядро с размери $M \times M$ можем да изчислим изходното изображение по следната формула:

$$g(i,j) = \sum_{m=-\frac{M}{2}}^{\frac{M}{2}} \sum_{n=-\frac{M}{2}}^{\frac{M}{2}} h(m,n) f(i-m, j-n)$$

В зависимост от съответното приложение съществуват различни ядра с размери и форма, които определят операцията. Най-важните от тях ще разгледаме по-долу.

За разлика от честотното пространство, в обектното пространство може да се реализира нелинеен филтър. В случая сумирането в конволюционната функция се заменя с някакъв вид нелинеен оператор

$$g(i,j) = O_{m,n} [h(m,n) f(i-m, j-n)]$$

За повечето нелинейни филтри всички елементи на $h(i,j)$ са 1. Често използван нелинеен оператор е медианата, която връща „средната“ от входните данни.

Линейни и нелинейни филтри

Нека операторът за обработка на изображения F , действащ върху две входни изображения A и B , извежда на изхода изображения C и D съответно. Ако операторът F е *линеен*, то:

$$F(a \times A + b \times B) = a \times C + b \times D,$$

където a и b са константи. На практика това означава, че всеки пиксел на изхода на *линеен* оператор е претеглена сума на множеството пиксели във входното изображение.

Нелинейни са всички останали оператори. Например бинаризиращият оператор е нелинеен, т.к. индивидуално всеки пиксел от двете изображения A и B може да е под прага, докато пикселът, който се получава като сумираме A и B може да е над прага. Аналогично операторът абсолютна стойност е нелинеен, т.к.:

$$|-1+1| \neq |-1| + |1|$$

Такъв е и операторът за експонента

$$\exp(1+1) \neq \exp(1) + \exp(1)$$

Филтрация по средна стойност (линеен филтър)

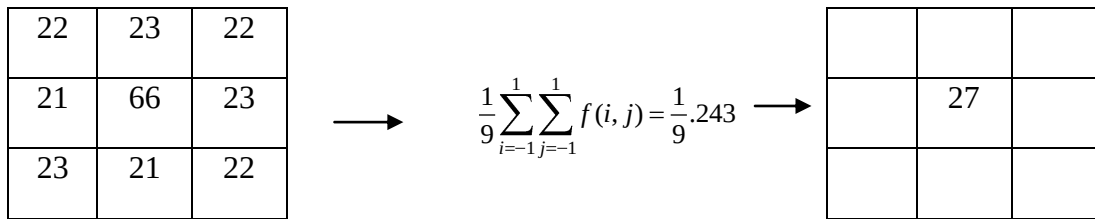
Изглаждането на изображение може да се постигне, ако стойността на всеки пиксел се замени със средната стойност от стойности на съседните му в рамките на апертура с размери $n \times n$. Апертурата определя геометрията и размерите на околното пространство спрямо даден пиксел (център). Обикновено се избира $n=3$ (по-рядко $n=5$). Изчислява се:

$$f(i, j) = S = \frac{1}{n^2} \sum_{r=-K}^K \sum_{l=-L}^L f(i+r, j+l),$$

където $K=L=(n-1)/2$, $i=2, 3, \dots, N-1$; $j=2, 3, \dots, M-1$ за $n=3$ или $i=3, 4, \dots, N-2$; $j=3, 4, \dots, M-2$ за $n=5$.

Филтрирането по средна стойност размива контурите на изображението, което е нежелателно при някои последващи етапи от обработката на изображението.

Пример:



Разновидност на филтрацията по средна стойност са *анизотропната* и *рекурентната* филтрации. Използват се изглаждащи матрици с размер $n \times n$. Примерни матрици могат да са:

$$A_1 = \frac{1}{10} \begin{vmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{vmatrix} \quad A_2 = \frac{1}{16} \begin{vmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{vmatrix}$$

Изчисляват се елементите на двумерния масив f' :

$$f'(i, j) = \sum_{r=-K}^K \sum_{l=-K}^K f(i+r, j+l) \cdot A(r, l),$$

където $K=1$; $i=2, 3, \dots, N-1$; $j=2, 3, \dots, M-1$. Ако в паметта на системата се запомнят изходната матрица g и резултатния масив f' , то се получава анизотропна филтрация. Ако изчислението се извършва чрез присвояването

$$f(i, j) = \sum_{r=-K}^K \sum_{l=-K}^K f(i+r, j+l) \cdot A(r, l),$$

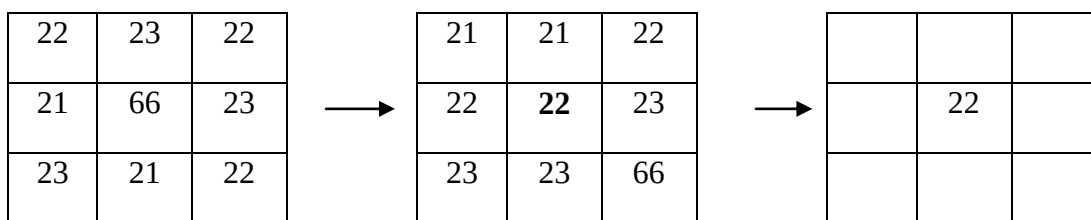
то се получава рекурентна филтрация. Новата стойност се запомня в съответния елемент на масива f . Съхранява се само един масив. Следователно рекурентната филтрация е за предпочитане, т.к. се използва по-малко памет.

Медианен филтър (нелинеен филтър)

При наличието на импулсен шум се използва *медианна* филтрация. Стойността на всеки елемент на изображението от входното поле се замества с медианата на стойностите в рамките на избраната апертура с размер $n \times n$ (обикновено $n=3$ или $n=5$):

$$f'(i, j) = MED(f(i+r, j+l)),$$

където $r, l = -(n-1)/2, \dots, (n-1)/2$. Елементите, попаднали в апертурата $n \times n$ се подреждат по стойност. Средният елемент (медианата) от това подреждане се присвоява на



Медианната филтрация запазва контурите, но „нащърбва“ ъглите.

Диференциални и интегриращи (изглаждащи) филтри

Нека с Φ означим функцията на даден филтър. Нека $\int \Phi(t) dt \approx \sum \Phi(t) = c$.
Диференциалните филтри се използват за оконтуряване.

$$\int \Phi(t) dt \approx \sum \Phi(t) = c \quad \left\{ \begin{array}{l} c > 1 - \text{интегриращ филтър с коефициент на усилване } c; \\ c == 1 - \text{интегриращ филтър}; \\ c == 0 - \text{диференциращ филтър}; \\ 0 < c < 1 - \text{интегриращ филтър с коефициент на отслабване } c; \\ c == -1 - \text{интегриращ филтър с инверсия}. \end{array} \right.$$

Изменението на яркостта на изображението определя неговото съдържание. Обикновено различните повърхнини, имат различна ориентация и коефициенти на отражение. На границите, между повърхнините се получава изменение на стойността на функцията на изображението $f(x, y)$, показваща интензитета на отразената светлина. Градиентът на функцията в точките на тези граници ще бъде голям. Стойностите на първата или втората производна на функцията на интензитета могат да се използват за определянето на ръбовете и границите на обектите в изображението.

Да разгледаме методи и оператори за отделяне на ръбове и граници, основаващи се на оценката на първата производна. За целта се извършва пространствено диференциране по x и y на функцията $f(x, y)$. Определя се модула на градиентните стойности за тази функция, които формират градиентното изображение $J(x, y)$,
 $J(x, y) = \|\Delta f(x, y)\| = \sqrt{J_x^2 + J_y^2} = \sqrt{(\partial f(x, y) / \partial x)^2 + (\partial f(x, y) / \partial y)^2}$. От градиентното изображение се преминава към неговото двоично представяне $b(x, y) = 1$, ако $J(x, y) \geq \theta$ или $b(x, y) = 0$, ако $J(x, y) < \theta$, където θ е праг. Обикновено стойността на прага се определя, анализирайки хистограмата на градиентното представяне $J(x, y)$, $x=1, 2, \dots, N$, $y=1, 2, \dots, M$.

Изображението се съхранява в паметта на компютъра посредством неговото дискретно (числово) представяне $g(i,j)$, $i \in 1, 2, \dots, N$, $j \in 1, 2, \dots, M$. Оценката на производните за функцията $g(i,j)$ се извършва посредством числови разлики, изпълнявани по определени схеми (едномерни, двумерни), наричани диференциални (градиентни) оператори:

Дискретно представяне	Диференциращ оператор	Стойност на градиента								
<table><tr><td>$g(i,j)$</td><td>$g(i,j+1)$</td></tr></table>	$g(i,j)$	$g(i,j+1)$	<table><tr><td>-1</td><td>+1</td></tr></table>	-1	+1	$J(i, j) = -g(i, j) + g(i, j+1) $				
$g(i,j)$	$g(i,j+1)$									
-1	+1									
<table><tr><td>$g(i,j)$</td></tr><tr><td>$g(i+1,j)$</td></tr></table>	$g(i,j)$	$g(i+1,j)$	<table><tr><td>+1</td></tr><tr><td>-1</td></tr></table>	+1	-1	$J(i, j) = g(i, j) - g(i+1, j) $				
$g(i,j)$										
$g(i+1,j)$										
+1										
-1										
<table><tr><td>$g(i,j)$</td><td>$g(i,j+1)$</td></tr><tr><td>$g(i+1,j)$</td><td>$g(i+1,j+1)$</td></tr></table>	$g(i,j)$	$g(i,j+1)$	$g(i+1,j)$	$g(i+1,j+1)$	<table><tr><td>+1</td><td>-1</td></tr><tr><td>+1</td><td>-1</td></tr></table>	+1	-1	+1	-1	$J(i, j) = g(i, j) - g(i+1, j+1) + g(i+1, j) - g(i, j+1) $
$g(i,j)$	$g(i,j+1)$									
$g(i+1,j)$	$g(i+1,j+1)$									
+1	-1									
+1	-1									

Последният диференциращ оператор от таблицата е предложен от **Робертс** и носи неговото име. Модулът на градиента (∇) се изчислява чрез дискретно пространствено диференциране посредством формулата:

$$J(i, j) = \sqrt{(g(i, j) - g(i+1, j+1))^2 + (g(i+1, j) - g(i, j+1))^2}$$

Смисълът на формулата е следният: ако точката (i, j) се намира в еднородна област, стойността на $J(i, j)$ ще бъде нула. Ако за редовете $i, i+1$ или пък за стълбовете $j, j+1$ се наблюдава изменението на интензитета, т.е. на функцията g , съответно $J(i, j)$ ще има достатъчно голяма стойност. За да се намали броят на изчисленията и повиши бързодействието (за сметка на точността) се използва следната значително по-проста формула: $J'(i, j) = |g(i, j) - g(i+1, j+1)| + |g(i+1, j) - g(i, j+1)|$. В качествено отношение функцията $J'(i, j)$ е също като $J(i, j)$ и нейните стойности са в интервала: $J(i, j) \leq J'(i, j) \leq \sqrt{2} J(i, j)$.

При използването на апертура с размер 2×2 , оценките на градиента се отнасят за апроксимираната междинна точка $(i+1/2, j+1/2)$, а не за пиксела (i,j) . Използването на апертура с размер 3×3 решава проблема с въвеждането на интерполирана точка, за която се оценява стойността на градиента. Такъв е **операторът на Собел**:

$$\Delta_x = \begin{vmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{vmatrix} \quad \Delta_y = \begin{vmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{vmatrix}$$

Формулите за оператора на Собел за следните: $\sqrt{J_x^2 + J_y^2}$, където $J_x = (g(i-1, j+1) + 2g(i, j+1) + g(i+1, j+1)) - (g(i-1, j-1) + 2g(i, j-1) + g(i+1, j-1))$, а

$J_y = (g(i-1, j-1) + 2g(i-1, j) + g(i-1, j+1)) - (g(i+1, j-1) + 2g(i+1, j) + g(i+1, j+1))$. За ускоряване на изчислението се използва формулата: $J(i, j) = |J_x| + |J_y|$.

Операторът на Собел (който между впрочем е нелинеен) позволява по-точна оценка на градиента на дискретната функция g от този на Робертс. От своя страна стойностите на J_x и J_y дават съответно информация за оценка на частните производни за функцията g в посока x и y . Ако изчисляваният градиент $|J|$ е достатъчно голям, то стръмнината и посоката на ръба може да се определи по формулата: $\alpha = \arctg(J_y / J_x)$.

Съществува следната зависимост за коефициентите a и b за оператора на Собел: $b/a=2.055$. Изхождайки от зависимостта могат да се дефинират множество маски на оператора. С нарастването на b се увеличава точността, но намалява чувствителността на оператора. В практиката се избира най-близкото цяло число до реалните коефициенти.

Операторът на Собел дава по-добри резултати в сравнение с подобните на него. Операторът на Собел се използва често при отделяне на ръбове и граници. Като недостатък на този оператор трябва да се отбележи, че при импулсни шумове в еднородни области, открива фалшиви ръбове.

Операторът на Лаплас се използва за оценка на втората производна. Двумерният Лапласиан на функцията $f(x,y)$ има вида: $\nabla^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$.

Диференциални лапласиан-оператори за дискретната функция $g(i, j)$ на основата на апертура 3×3 за четири или осем съседа се записват, съответно, по следния начин:

$$\nabla^2 \approx \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \nabla^2 \approx \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$J(i, j) = \frac{1}{4} (g(i-1, j) + g(i, j-1) + g(i, j+1) + g(i+1, j) - 4g(i, j))$$

$$J(i, j) = \frac{1}{8} (g(i-1, j-1) + g(i-1, j) + g(i-1, j+1) + g(i, j-1) + g(i, j+1) + g(i+1, j-1) + g(i+1, j) + g(i+1, j+1) - 8g(i, j))$$

Методите за получаване на градиентното изображение, използващи втората производна са чувствителни към шумове в изображението и ги усилват, тъй като дори много малки локални пикове на първата производна водят до пресичания на нулата (zero-crossing) за втората производна (стойността на втората производна за функцията $f(x,y)$, чиято стойност е нула при изменението на интензитета). Налага се редуциране на шумовете преди прилагането на тези оператори.

Литература: 1. Лекция „Филтри в обектното пространство“ (2k9,2k12), доц. Д. Димов;
2. Гочев Г.: Компютърно зрение и невронни мрежи. ТУ-София, 1998.

От автора: Темата е развивана предимно от т. 1 и т. 2 (вкл. и интернет за по-добро разяснение на някои от концепциите). Възможно е да има неточности и разминавания между някои от нещата т. 1 и т. 2.

Моля за извинение за допуснати грешки и неточности, но темата е развивана предимно по малките часове на нощта, а и не е редактирана.