1 Vaizdų vidurkinimas ir požymių išskyrimas

1.1 Glodus vienmatis eksponentinis filtras

Apibrėšime eksponentinį tolydų kintamojo x filtrą formule

$$v_\sigma(x) = \left(1 + \frac{|x|}{\sigma}\right) e^{-\frac{|x|}{\sigma}}, x \in (-\infty, \infty). \tag{1}$$

Čia σ yra filtro parametras. Kad filtro reikšmės gestų begalybėje, reikalausime, kad $\sigma > 0$. Nesunku patikrinti, kad su bet kokia fiksuota parametro $\sigma > 0$ reikšme filtras yra du kartus tolydžiai diferencijuojamas atžvilgiu kintamojo x. Todėl šį filtrą vadinsime *glodžiu eksponentiniu filtru*.

Bet kokiam signalui $u = u(x)$, filtravimo operacija apibrėžiama formule

$$(u \star v_\sigma)(x) = \int_{-\infty}^{\infty} u(y) v_\sigma(x-y) dy = \int_{-\infty}^{\infty} v_\sigma(y) u(x-y) dy. \tag{2}$$

Skaitmeninių signalų atveju laikoma, kad

$$u(x) = \sum_{n=-\infty}^{\infty} u_n \delta(x-n), \tag{3}$$

kur $\delta = \delta(x)$ yra delta-dirako funkcija pasižyminti savybe

$$\int_{-\infty}^{\infty} \delta(y) \phi(y) dy = \phi(0) \tag{4}$$

visoms glodžioms integruojamoms funkcijoms $\phi = \phi(x)$.Pasinaudoję (3) ir (4) lygybėmis, gauname tokią filtravimo operacijos išraišką:

$$(u \star v_\sigma)_n = \sum_{m=-\infty}^{\infty} u_m v_\sigma(n-m) = \sum_{m=-\infty}^{\infty} v_\sigma(m) u_{n-m}, n = 0, \pm 1, \pm 2, \cdots. \tag{5}$$

Atlikime filtravimo operaciją vienetiniam skaitmeniniam signalui, t. y. signalui

$$u(x) = \sum_{n=-\infty}^{\infty} \delta(x-n).$$

Pasinaudoje filtravimo formulę, visiems sveikiesiems n, gausime

$$(u \star v_\sigma)_n = \sum_{m=-\infty}^{\infty} v_\sigma(n-m) = \sum_{m=-\infty}^{\infty} v_\sigma(m) = \sum_{m=-\infty}^{\infty} (1 + \frac{|m|}{\sigma}) e^{-\frac{|m|}{\sigma}}.$$

Pritaikę geometrinės progresijos sumos formulę rasime

$$\sum_{m=-\infty}^{\infty} e^{-\frac{|m|}{\sigma}} = \frac{1+e^{\frac{-1}{\sigma}}}{1-e^{\frac{-1}{\sigma}}}.$$

Kadangi

$$\sum_{m=0}^{\infty} m e^{mt} = (\sum_{m=0}^{\infty} e^{mt})'_t = \frac{e^t}{(1-e^t)^2},$$

tai galutinai gauname

$$(u \star v_\sigma)_n = \sum_{m=-\infty}^{\infty} (1+\frac{|m|}{\sigma}) e^{-\frac{|m|}{\sigma}} = \frac{\sigma(1-e^{\frac{-2}{\sigma}}) + 2e^{\frac{-1}{\sigma}}}{\sigma(1-e^{\frac{-1}{\sigma}})^2} = \frac{1}{q(\sigma)}. \tag{6}$$

Modifikuosime glodaus eksponentinio filtro apibrėžimą taip, kad jo atsakas į vienetinį skaitmeninį signalą būtų lygus 1. Kad tai pasiekti pakanka (1) dešiniąją pusę padauginti iš normuojančio daugikli $q(\sigma)$ apibrėžto formule (6). Tokiu būdu gauname tokią normuotą eksponentinio filtro išraišką:

$$v_\sigma(x) = \frac{\sigma(1-e^{\frac{-1}{\sigma}})^2}{\sigma(1-e^{\frac{-2}{\sigma}}) + 2e^{\frac{-1}{\sigma}}} \left(1+\frac{|x|}{\sigma}\right) e^{-\frac{|x|}{\sigma}}, x \in (-\infty, \infty). \tag{7}$$

Keičiant eksponentinio filtro parametrą σ galima pasiekti norimą signalo vidurkinimo lygį. Kuo σ arčiau nuliui, tuo filtravimo operacija mažiau glodina filtruojamą signalą ir atvirkščiai didinant parametro σ reikšmę didiname vidurkinimo laipsnį.

Išsiaiškinsime kaip efektyviai atlikti skaitmeninio signalo vidurkinimą eksponentiniu filtru. Realių skaitmeninių signalų žinomų skaitmeninių reikšmių kiekis būna baigtinis. Tarkime žinome

$$u_0, u_1, u_2, \cdots, u_{N-1}$$

reikšmes. Laikysime, kad filtruojamo signalo kraštinės reikšmės u_0 ir u_{N-1} pratęstos tolydžiai, t. y.

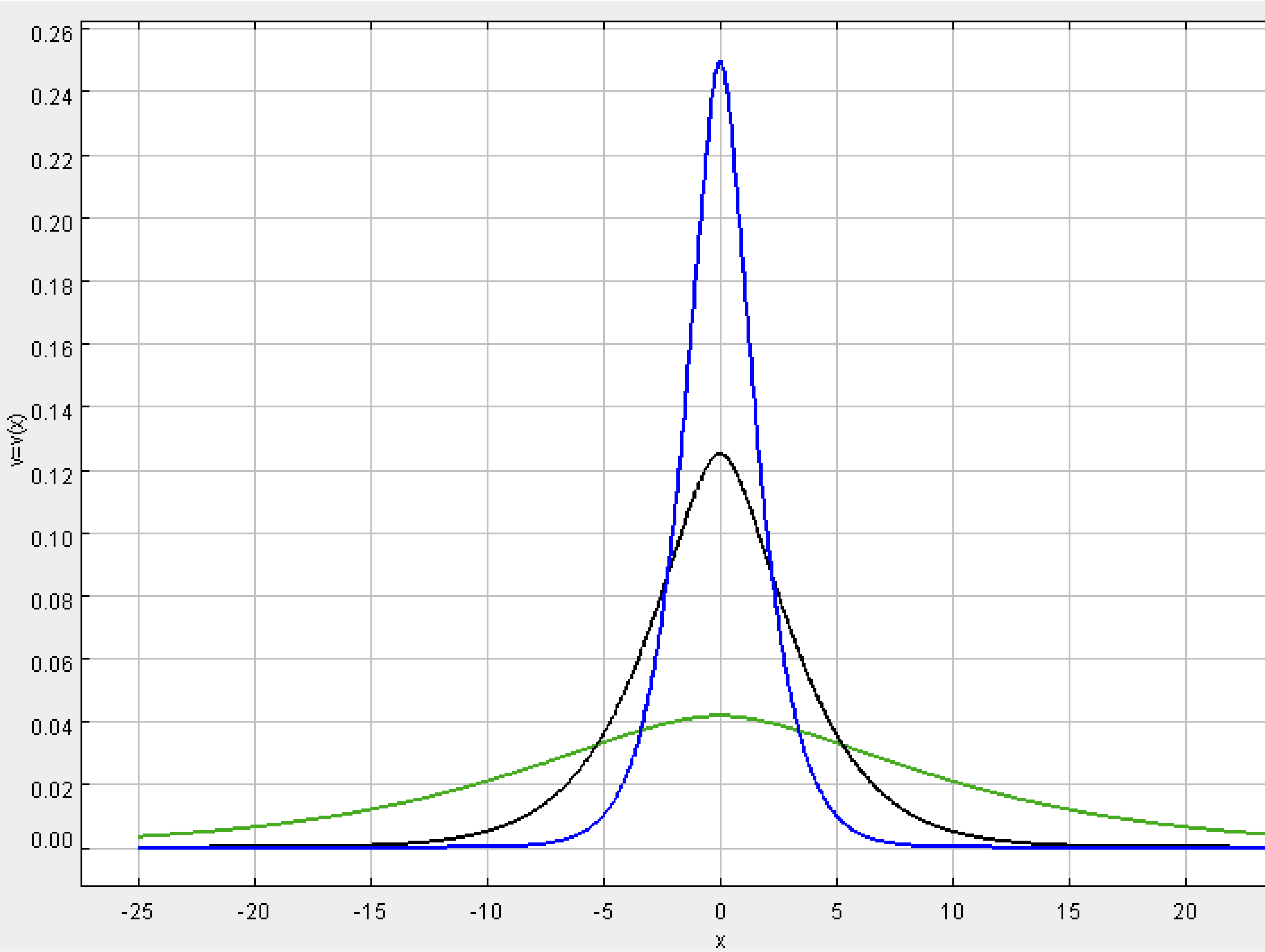
$$u_n \equiv u_0, \text{ kai } n < 0$$

ir

$$u_n \equiv u_{N-1}, \text{ kai } n \geq N.$$

Kad supaprastinti formules, pradžioje nerašysime normuojančio daugiklio. Išskaidysime filtravimo operacijos sumą į dvi dalis:

$$(u \star v_\sigma)_n = \sum_{m=-\infty}^{\infty} v_\sigma(m) u_{n-m} = -u_n + \sum_{m=0}^{\infty} (1+\frac{m}{\sigma}) e^{\frac{-m}{\sigma}} u_{n-m} + \sum_{m=0}^{\infty} (1+\frac{m}{\sigma}) e^{\frac{-m}{\sigma}} u_{n+m}.$$



1 paveikslėlis: Glodžių eksponentinių filtrų v_σ grafikai. Mėlynas $\sigma = 1$, juodas $\sigma = 2$, žalias $\sigma = 8$

Užrašydami išskaidymo formulę pasinaudojome filtro simetriškumu, t. y. savybe $v_\sigma(x) = v_\sigma(-x)$. Pradžioje raskime

$$(u \star v0^+_\sigma)_n = \sum_{m=0}^{\infty} e^{\frac{-m}{\sigma}} u_{n-m}. \tag{8}$$

Kadangi visiems neigiamiems indeksams $u_{n-m} = u_0$, tai

$$(u \star v0^+_\sigma)_0 = \frac{1}{1 - e^{\frac{-1}{\sigma}}} u_0.$$

Teigiamiems indeksams $n > 0$ galime užrašyti rekurentinį ryšį

$$(u \star v0^+_\sigma)_n = e^{\frac{-1}{\sigma}} (u \star v0^+_\sigma)_{n-1} + u_n.$$

Dabar apskaičiuokime

$$(u \star v1_\sigma^+)_n = \sum_{m=0}^{\infty} m e^{\frac{-m}{\sigma}} u_{n-m}. \tag{9}$$

Dėl kairiosios kraštinės sąlygos gauname

$$(u \star v1_\sigma^+)_0 = \frac{e^{\frac{-1}{\sigma}}}{(1 - e^{\frac{-1}{\sigma}})^2} u_0,$$

o teigiamiems indeksams $n > 0$ galime užrašyti sąryšį

$$\begin{aligned}(u \star v1_\sigma^+)_n &= \sum_{m=0}^{\infty} m e^{\frac{-m}{\sigma}} u_{n-m} = \sum_{m=0}^{\infty} (m+1) e^{\frac{-m-1}{\sigma}} u_{n-1-m} \\ &= e^{\frac{-1}{\sigma}}((u \star v1_\sigma^+)_{n-1} + (u \star v0_\sigma^+)_{n-1}), n = 1, 2, \cdots, N-1. \end{aligned} \tag{10}$$

Analogiškai raskime

$$(u \star v0_\sigma^-)_n = \sum_{m=0}^{\infty} e^{\frac{-m}{\sigma}} u_{n+m}. \tag{11}$$

Kadangi visiems indeksams $m \geq 0$ $u_{N-1+m} = u_{N-1}$, tai

$$(u \star v0_\sigma^-)_{N-1} = \frac{1}{1 - e^{\frac{-1}{\sigma}}} u_{N-1}.$$

Likusiems indeksams n galime užrašyti sąryšį

$$(u \star v0_\sigma^-)_n = e^{\frac{-1}{\sigma}} (u \star v0_\sigma^-)_{n+1} + u_n.$$

Liko apskaičiuoti

$$(u \star v1_\sigma^-)_n = \sum_{m=0}^{\infty} m e^{\frac{-m}{\sigma}} u_{n+m}. \tag{12}$$

Dėl dešiniosios kraštinės sąlygos gauname

$$(u \star v1_\sigma^-)_{N-1} = \frac{e^{\frac{-1}{\sigma}}}{(1 - e^{\frac{-1}{\sigma}})^2} u_{N-1},$$

o kitiems indeksams $n \geq 0$ galioja sąryšis

$$\begin{aligned}(u \star v1_\sigma^-)_n &= \sum_{m=0}^{\infty} m e^{\frac{-m}{\sigma}} u_{n+m} = \sum_{m=0}^{\infty} (m+1) e^{\frac{-m-1}{\sigma}} u_{n+1+m} \\ &= e^{\frac{-1}{\sigma}}((u \star v1_\sigma^-)_{n+1} + (u \star v0_\sigma^-)_{n+1}), n = N-2, N-3, \cdots, 0. \end{aligned} \tag{13}$$

Galutinės filtravimo formulės su normuojančiu daugikliu atrodo taip:

$$(u \star v_\sigma)_n = q(\sigma)((u \star v0_\sigma^+)_n + \frac{(u \star v1_\sigma^+)_n}{\sigma} + (u \star v0_\sigma^-)_n + \frac{(u \star v1_\sigma^-)_n}{\sigma} - u_n). \tag{14}$$

1.2 Filtruoto vienmačio signalo išvestinės

Pateiktos formulės tinka pradinio skaitmeninio signalo greitam glodinimui. Kad apskaičiuoti $v0_n^+$, $v0_n^-$, $v1_n^+$ ir $v1_n^-$ reikia vienos daugybos ir vienos sudėties, o galutiniam normuotam filtravimo rezultatui gauti reikia atlikti dar dvi daugybas ir tris sudėties/atimties operacijas, taigi mūsų filtravimo algoritmo sudėtingumas yra 6 daugybos ir 7 sudėties operacijos vienam taškeliui. Tačiau praktikoje dažnai reikia ne tik suglodinti pradinį signalą bet ir įvertinti jo pirmąją ir antrąją išvestines. Išvestinės dažniausiai yra aproksimuojamos pirmosios ir antrosios eilės skirtumais. Tačiau mūsų atveju galima apskaičiuoti tiksliai filtruoto signalo išvestines. Formaliai diferencijuodami gauname

$$(u \star v_\sigma)'(x) = \int_{-\infty}^{\infty} u(y)v_\sigma'(x-y)dy = \frac{-q(\sigma)}{\sigma^2}\int_{-\infty}^{\infty} u(y)(x-y)e^{-\frac{|x-y|}{\sigma}}dy. \tag{15}$$

Diskrečiu atveju tiksli filtruoto signalo išvestinės formulė su normuojančiu daugikliu atrodo taip:

$$(u \star v_\sigma)_n' = \frac{-q(\sigma)}{\sigma^2}\sum_{m=-\infty}^{\infty} m e^{-\frac{|m|}{\sigma}} u_{n-m} = \frac{-q(\sigma)}{\sigma^2}((u \star v1_\sigma^+)_n - (u \star v1_\sigma^-)_n). \tag{16}$$

Iš (16) formulės matome, kad išsaugojus tarpinius glodinimo filtravimo rezultatus, išvestines apskaičiavimo papildomos sąnaudos yra minimalios, tiksliau, kad apskaičiuoti filtruoto skaitmeninio signalo išvestinę viename taškelyje n reikia atlikti vieną sudėties ir daugybos operaciją.

Analogiškai rasime tikslią antrosios eilės išvestinės formulę. Formaliai diferencijuodami du kartus filtravimo rezultatą, gauname

$$(u \star v_\sigma)''(x) = \int_{-\infty}^{\infty} u(y)v_\sigma''(x-y)dy = \frac{-q(\sigma)}{\sigma^2}\int_{-\infty}^{\infty} u(y)(1-\frac{|x-y|}{\sigma})e^{-\frac{|x-y|}{\sigma}}dy. \tag{17}$$

Diskrečiu atveju filtruoto signalo antrosios eilės išvestinės formulė yra tokia:

$$\begin{aligned}(u \star v_\sigma)_n'' &= \frac{-q(\sigma)}{\sigma^2}\sum_{m=-\infty}^{\infty}(1-\frac{|m|}{\sigma})e^{-\frac{|m|}{\sigma}}u_{n-m}\\ &= \frac{-q(\sigma)}{\sigma^2}((u \star v0_\sigma^+)_n - \frac{(u \star v1_\sigma^+)_n}{\sigma} + (u \star v0_\sigma^-)_n - \frac{(u \star v1_\sigma^-)_n}{\sigma} - u_n). \end{aligned}\tag{18}$$

Taigi vėl gauname ekonomišką tikslų antrosios eilės išvestinės apskaičiavimo algoritmą, kurio sudėtingumas yra 2 daugybos ir 4 sudėties/atimties operacijos.

Reikia pastebėti, kad jei mūsų filtras nebūtų du kartu tolydžiai diferencijuodamas, tai formalus išvestinių įkėlimas po konvoliucijos (filtravimo) integralų ženklu duotų klaidingas formules. Tačiau net ir su glodžiu filtru gauname, kad konstantos antrosios eilės išvestinė yra nenulinė, o etaloninių signalų $u_n = n$ ir $u_n = n^2$ vidurkintos reikšmės ir jų išvestinės tiksliai nesutampa su laukiamomis reikšmėmis. Kad ištaisyti šias filtrų ydas, reikalausime, kad visi filtrai būtų apskaičiuotų pagal (8), (9), 11 ir 12 išraiškas funkcijų tiesiniai dariniai, vidurkintas signalas nekeistų vienetinį ($u_n = 1$)ir kvadratinį ($u_n = n^2$) signalus, tiesinio signalo ($u_n = n$) išvestinė būtų lygi vienetiniam signalui ir ir kvadratinio signalo $u_n = n^2$ išvestinė būtų lygi konstantai 2. Pagal šias sąlygas gauname tokias galutines formules:

$$
\begin{aligned}
\overline{u}_n &= a_0((u \star v0_\sigma)_n - u_n) + b_0(u \star v1_\sigma)_n, \text{ (vidurkintas signalas)} &(19)\\
\overline{u}'_n &= a_1(u \star v1^+_\sigma)_n + b_1(u \star v1^-_\sigma)_n, \text{ (vidurkinto signalo išvestinė)} &(20)\\
\overline{u}''_n &= a_2((u \star v0_\sigma)_n - u_n) + b_2(u \star v1_\sigma)_n, \text{ (antrosios eilės išvestinė)}\ , &(21)
\end{aligned}
$$

kur

$$
\begin{aligned}
v0_{\sigma n} &= v0^+_{\sigma\, n} + v0^-_{\sigma\, n}, &(22)\\
v1_{\sigma n} &= v1^+_{\sigma\, n} + v1^-_{\sigma\, n}, &(23)\\
q^0(\sigma) &= \sum_{m=-\infty}^{\infty} e^{-\frac{|m|}{\sigma}} = (1 + e^{\frac{-1}{\sigma}})/(1 - e^{\frac{-1}{\sigma}}), &(24)\\
q^1(\sigma) &= \sum_{m=-\infty}^{\infty} |m| e^{-\frac{|m|}{\sigma}} = 2 * e^{\frac{-1}{\sigma}}/(1 - e^{\frac{-1}{\sigma}})^2, &(25)\\
q^2(\sigma) &= \sum_{m=-\infty}^{\infty} m^2 e^{-\frac{|m|}{\sigma}} = 2e^{\frac{-1}{\sigma}}(1 + e^{\frac{-1}{\sigma}})/(1 - e^{\frac{-1}{\sigma}})^3, &(26)\\
q^3(\sigma) &= \sum_{m=-\infty}^{\infty} |m|^3 e^{-\frac{|m|}{\sigma}} = 2e^{\frac{-1}{\sigma}}(1 + 4 * e^{\frac{-1}{\sigma}} + e^{\frac{-2}{\sigma}})/(1 - e^{\frac{-1}{\sigma}})^4, &(27)
\end{aligned}
$$

o daugikliai a ir b randami iš lygčių

$$\begin{aligned}
a_0 q^0(\sigma) + b_0 q^1(\sigma) &= 1, && (28)\\
a_0 q^2(\sigma) + b_0 q^3(\sigma) &= 0, && (29)\\
a_1 q^2(\sigma) + b_1 0 &= -1, && (30)\\
a_1 0 + b_1 * q^2(\sigma) &= 1, && (31)\\
a_2 q^0(\sigma) + b_2 q^1(\sigma) &= 0, && (32)\\
a_2 q^2(\sigma) + b_2 q^3(\sigma) &= 2.. && (33)
\end{aligned}$$

Alternatyvus koeficentų a_0 ir b_0 apibrėžimo variantas:

$$\begin{aligned}
a_0 &= \frac{a_2}{a_2 q^0 - b_2 q^1}, && (34)\\
b_0 &= \frac{-b_2}{a_2 q^0 - b_2 q^1}. && (35)
\end{aligned}$$

Skaičiuojant a_0 ir b_0 pagal šias formules vidurkinta konstanta išlieka ta pačia konstanta ir išlaikoma simetrija: $a_0 = \text{const}\ a_2$, $b_0 = \text{const}\ b_2$.

1.2.1 Praktinė vienmačių signalų vidurkinimo užduotis

Parašyti programą, kuri apskaičiuotų duoto skaitmeninio signalo suglodinimą eksponentiniu filtru ir rastų suglodinto signalo pirmąsias ir antrąsias išvestines. Algoritmo testavimui imkite tokius duomenis:

$$u_n = \begin{cases} -1, & kai \quad n < 100, \\ 3, & kai \quad n < 200, \\ 2, & kai \quad n < 300, \end{cases}$$

$$\sigma = 2.$$

Dumenys kiekvieno algoritmo žingsnio testavimui: $\sigma = \frac{1}{\log 2}$,

$$\begin{aligned}
u &= 64 \quad 32 \quad 64 \quad 128,\\
(u \star v0_\sigma^+) &= 128 \quad 96 \quad 112 \quad 184,\\
(u \star v1_\sigma^+) &= 128 \quad 128 \quad 112 \quad 112,\\
(u \star v0_\sigma^-) &= 128 \quad 128 \quad 192 \quad 256,\\
(u \star v1_\sigma^-) &= 176 \quad 224 \quad 256 \quad 256.
\end{aligned}$$

$a_0 = \frac{13}{27}$, $b_0 = \frac{-1}{9}$, $a_1 = \frac{-1}{12}$, $b_1 = \frac{1}{12}$ $a_2 = \frac{-2}{27}$, $b_2 = \frac{1}{18}$.

$$\begin{array}{rcllll}
\overline{u} & = & \frac{13}{27}192 - \frac{1}{9}304 & \frac{13}{27}192 - \frac{1}{9}352 & \frac{13}{27}240 - \frac{1}{9}368 & \frac{13}{27}312 - \frac{1}{9}368, \\
\overline{u} & = & \frac{176}{3} & \frac{160}{3} & \frac{223}{3} & \frac{327}{3}, \\
\overline{u}' & = & 4 & 8 & 12 & 12, \\
\overline{u}'' & = & \frac{8}{3} & \frac{15}{3} & \frac{7}{3} & \frac{-8}{3}.
\end{array}$$

Atlikite filtravimą etaloniniams duomenis $u_n = 1$, $u_n = n$ ir $u_n = n^2$ ir pakomentuokite gautus filtravimo rezultatus.

Šioje ir kitose programose nebūtina realizuoti vartotojo grafinę sąsają.

Vertinama iki 0.75 egzamino balų. Jei dėl nepateisinamų priežasčių atsiskaitoma vėliau nei keturios savaitės nuo pirmųjų pratybų, už kiekvienos savaitės vėlavimą vertinimas mažinamas 0.25 balo.

1.3 Vaizdų vidurkinimas ir gradiento įvertis

Apibendrinsime vienmačių signalų eksponentinį vidurkimo filtrą (1) dvimačiams vaizdams:

$$v_\sigma(x, y) = \left(1 + \frac{|x|}{\sigma}\right)\left(1 + \frac{|y|}{\sigma}\right) e^{-\frac{|x|+|y|}{\sigma}}, x, y \in (-\infty, \infty). \tag{36}$$

Nesunku pastebėti, kas dvimatis filtras yra vienmačių filtrų sandauga, t. y.

$$v_\sigma(x, y) = v_\sigma(x) v_\sigma(y).$$

Ši filtro savybė lemia, kad dvimatis filtravimas gali būti atliktas pirma vaizdo eilutėms taikant eksponentinį filtravimą, o vėliau gautam filtruotam vaizdui atliekamas eksponentinis filtravimas stulpeliams. Lygiai tą patį rezultatą galima pirma atliekant filtravimą vaizdo stulpeliams, o paskui eilutėms. Matematiškai ši savybė ekvivalenti teiginiui, kad kartotinio integralo reikšmė nepriklauso nuo integravimo tvarkos, jei integruojama funkcija yra pakankamai gera. Koks yra sąvokos pakankamai gera tikslus matematinis turinys mes netikslinsime, nes realaus pasaulio analoginiai vaizdai yra tolydūs, mūsų filtrai taip pat tolydūs, tai tolydžios funkcijos (vaizdo) ir tolydaus filtro sandauga taip pat yra tolydi funkcija ir tokios kartotinio integralo reikšmės nepriklauso nuo integravimo tvarkos. Todėl diskrečiu atveju vidurkinant vaizdą reikia pritaikyti (19) filtrą eilutėms, o po to pritaikyti tą patį filtrą gautam vidurkintam vaizdui. Atliekant šias operacijas galima naudoti skirtingų σ filtrus, bet dėl filtro simetriškumo prasminga naudoti tą pačią σ.

2 paveikslėlyje iliustruoti pradinį rainelės vaizdą, jo vidurkintą, gradiento ir Laplaso vaizdus. Kad gradiento ir Laplaso vaizdai būtų labiau informatyvūs, iš gautų vaidų modulio buvo ištraukta kvadratinė šaknis ir padauginta iš ženklo. Vidurkinimo skalės parametro reikšmė $\sigma = 3$. Kaip gaunamas vidurkintas vaizdas mes jau išsiaiškinome. Gradiento vaizdas žymi vidurkinto vaizdo gradiento modulį, t. y.

$$|\nabla \overline{u}| = \sqrt{(\frac{\partial \overline{u}}{\partial x})^2 + (\frac{\partial \overline{u}}{\partial y})^2}, \tag{37}$$

o vidurkinto vaizdo Laplasas randamas pagal tokią taisyklę:

$$\Delta \overline{u} = \frac{\partial^2 \overline{u}}{\partial x^2} + \frac{\partial^2 \overline{u}}{\partial y^2}. \tag{38}$$

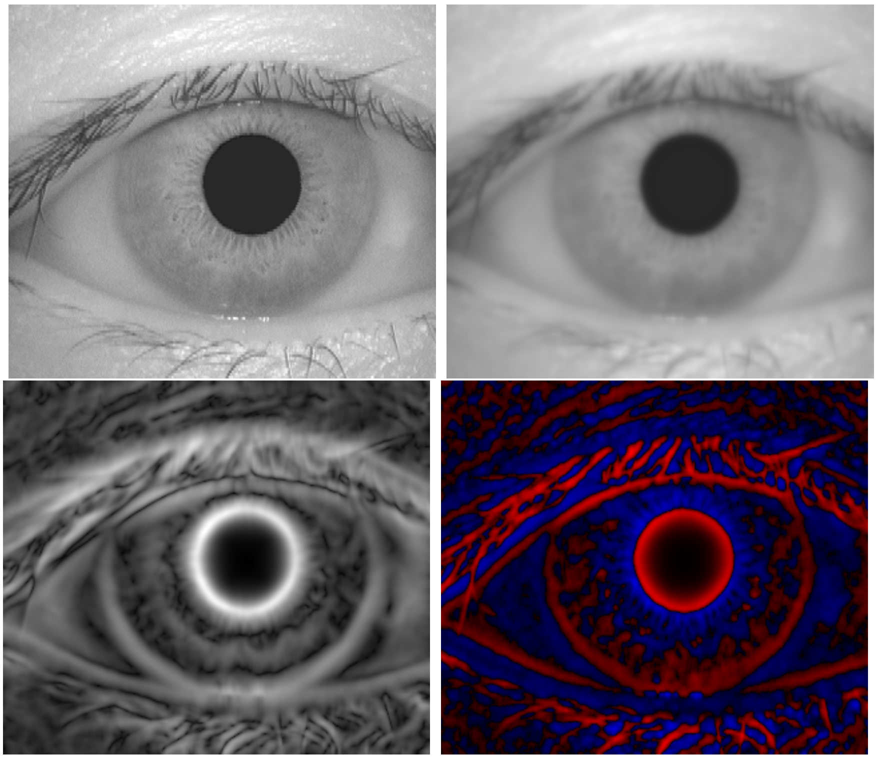
Kadangi Laplaso vaizdo reikšmės gali būti neigiamos ir teigiamos, teigiamos reikšmės vaizduojamos raudonai, neigiamos mėlynai. Kad rasti vaizdo išvestines x kryptimi, reikia taikyti (20) ir (21) taisykles eilutėms ir (19) taisyklę stulpeliams. Vidurkinto vaizdo išvestinės y kryptimi randamos taikant (20) ir (21) taisykles stulpeliams ir (19) taisyklę eilutėms. Kadangi pratybų metu vėlesnėse užduotyse pasirinktinai lyginsite pirštų atspaudus, veidus arba raineles, pateiksime ir veido bei rainelių vidurkintų vaizdų pavyzdžius. Iš pateiktų vidurkinto veido, gradiento ir Laplaso vaizdų matyti, kad ir gradientas ir Laplaso vaizdai atrodo estetiškai, todėl yra prasminga iš jų išskirti požymius.

Vidurkinti piršto atspaudo gradiento ir Laplaso vaizdai atrodo neestetiškai, nes susidaro įspūdis, kad piršto atspaudo linijų tankis yra padvigubėjęs du kartus. Todėl vėliau išskiriant požymius rekomenduojama naudoti tik vidurkintą piršto atspaudą arba Laplaso vaizde išskirti apibendrintus minimumo ir maksimumo taškus.

1.3.1 Dvimačio vidurkinimo praktinė užduotis

Parašyti programą, kuri apskaičiuotų duoto skaitmeninio vaizdo suglodinimą eksponentiniu simetriniu filtru ir rastų suglodinto vaizdo gradiento modulio ir Laplaso vaizdus.

Algoritmo testavimui imkite $\sigma = 3$ ir tokius vaizdus:



2 paveikslėlis: Casia 1 duomenų bazės pirmosios rainelės pradinis vaizdas (kairėje viršuje), jos vidurkintas analogas (dešinėje viršuje, $\sigma = 3$, vidurkinto vaizdo gradiento modulis (kairėje apačioje) ir vidurkinto vaizdo Laplasas (dešinėje apačioje)

- Casia 1 DB pirmoji rainelė *figures/001_1_1.bmp*
- Ferret grey DB pirmasis veidas *figures/00001da010_930831.png*
- FVC 2000 DB1 pirmasis piršto atspaudas *figures/1_1.tif*

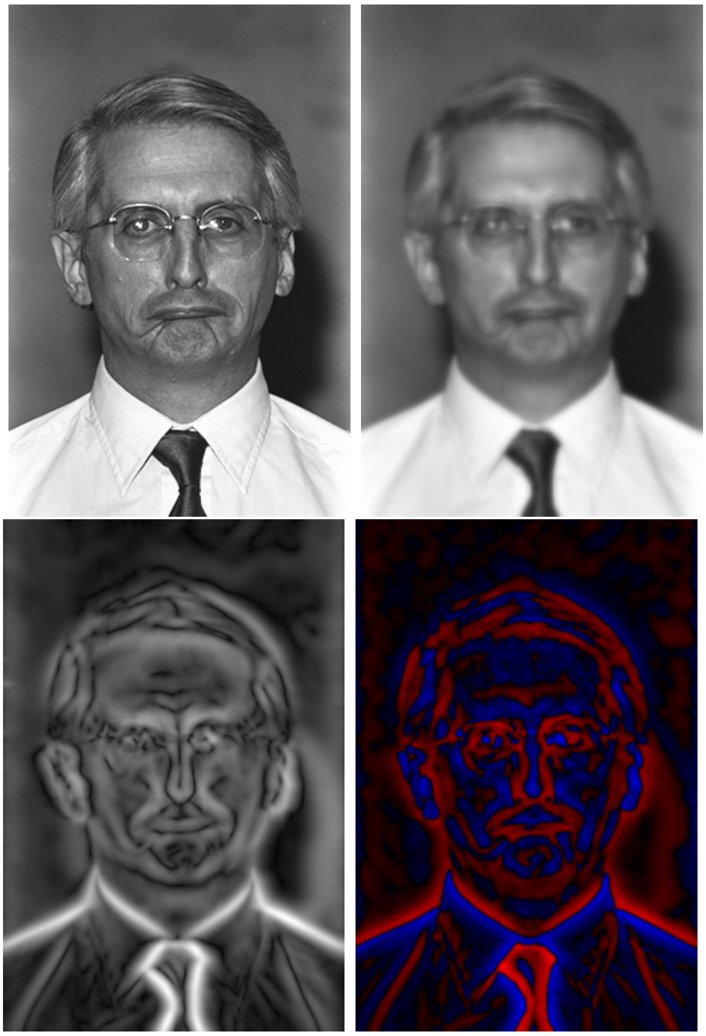
Jei kyla problemų su kokio nors formato duomenų nuskaitymu, galite juos konvertuoti į jums patogų formatą panaudojant http://www.irfanview.com/ ir pan. priemones..

Kontrolinių vaizdų apdorojimo rezultatus galite palyginti su šių konspektų iliustracijomis; turi gautis vaizdai panašūs į 2-4 paveikslėlių vaizdus.

Vertinama iki 0.75 egzamino balų. Jei dėl nepateisinamų priežasčių atsiskaitoma vėliau nei penkios savaitės nuo pirmųjų pratybų, už kiekvienos savaitės vėlavimą vertinimas mažinamas 0.25 balo.

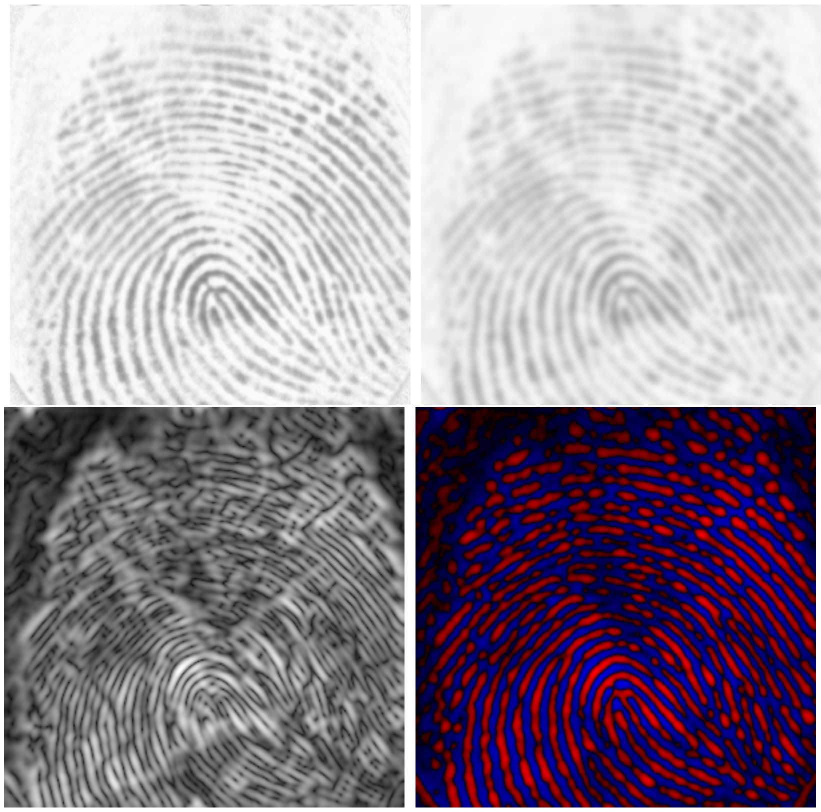
1.4 Vaizdų požymiai

Vaizdų požymiams įvertinti naudosime gradiento modulio apibendrintus lokalius ekstremumus ir Laplaso nulio kirtimus [1]. Abudu šie požymiai apytikriai išskiria



3 paveikslėlis: Ferret duomenų bazės veido pradinis vaizdas (kairėje viršuje), jo vidurkintas analogas (dešinėje viršuje, $\sigma = 3$, vidurkinto vaizdo gradiento modulis (kairėje apačioje) ir vidurkinto vaizdo Laplasas (dešinėje apačioje)

vaizdo objektų kontūrus. Tačiau dėl išvestinių nestabilumo dalis gaunamų kontūrų neturi aiškios interpretacijos. Kad sumažinti tokių kontūrų skaičių galima didinti vidurkinimo mastelio σ vertę. Tačiau toks būdas mažina realių kontūrų padėties nustatymo tikslumą. Kitas būdas derinti gradiento apibendrintų lokalių maksimumų ir Laplaso nulio kirtimų kontūrus. Jei ir vienu ir kitu būdu gaunami greta du tokie kontūrai, tai labai padidina tikimybę, kad jie atitinka realų kontūrą. Dar vienas būdas yra derinti kontūrus gautus vidurkinant vaizdą su keliais skirtingais σ. Čia vėl yra ieškomi artimi keliuose vidurkintuose vaizduose kontūrai, tačiau tiksliai apibrėžti kontūrų artumo sąvoką gana keblu. Gana neblogai ir paprastai netikri kontūrai eliminuojami pasirenkant slenkstį ir ignoruojant kontūrus, kurių gradiento



4 paveikslėlis: FVC 2000 DB1 duomenų bazės pirmojo piršto atspaudo pradinis vaizdas (kairėje viršuje), jo vidurkintas analogas (dešinėje viršuje, $\sigma = 3$, vidurkinto vaizdo gradiento modulis (kairėje apačioje) ir vidurkinto vaizdo Laplasas (dešinėje apačioje)

modulio reikšmės mažesnės už slenkstį ir eliminuojant trumpus kontūrus. Tačiau ir vienu ir kitu atveju reikia turėti slenkstį nusakantį gradiento arba kontūro ilgio mažumą, o tai menkina tokio būdo vertę.

Kokiu būdu išskirti gradientiniame arba Laplaso vaizde kontūrus? Pradėsime nuo Laplaso vaizdo, kadangi jame kontūrai išskiriami remiantis paprastu nulių kirtimo principu, kuris pažymi diskrečias pozicijas, kurių kaimynystėje Laplasas lygus nuliui. Labai maža tikimybė, kad kokioje nors diskrečioje pozicijoje Laplasas bus tiksliai lygus nuliui. Tačiau kadangi Laplaso vaizde matome ir raudonas teigiamo ženklo zonas ir mėlynas neigiamo ženklo zonas, tai aišku, kad atsiras jų jungimosi taškus. Paprasčiausias būdas rasti tokius taškus yra fiksuoti poziciją (m, n) ir patikrinti sąlygą

$$\Delta\overline{u}_{m,n+1} * \Delta\overline{u}_{m,n-1} \leq 0 \mid \Delta\overline{u}_{m+1,n} * \Delta\overline{u}_{m-1,n} \leq 0.$$

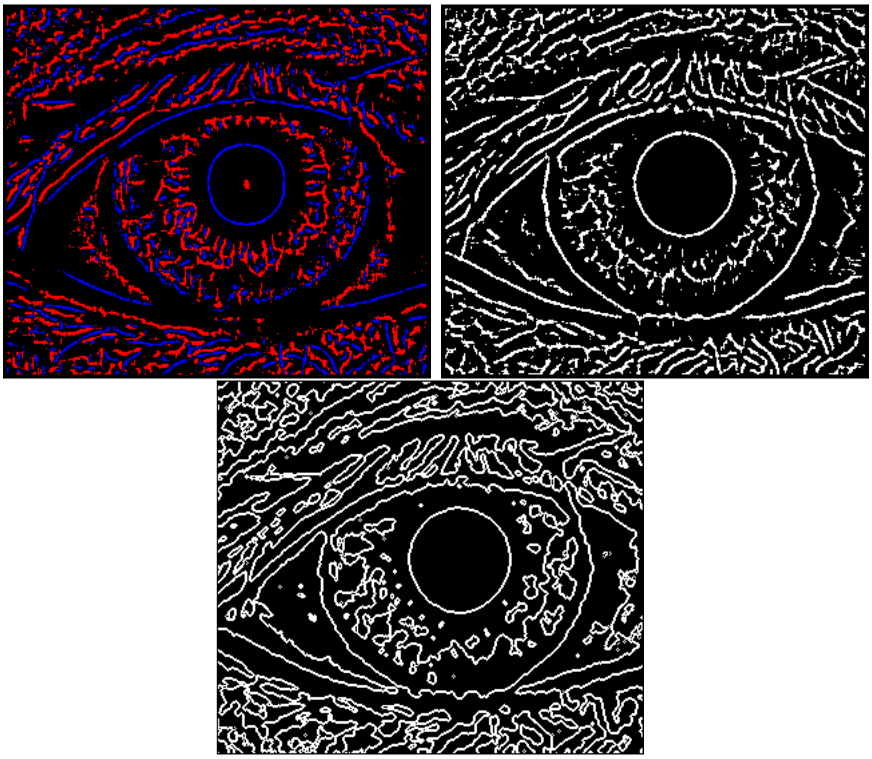
Jei Laplaso vaizde randame tokį taškelį, jį pažymime tarkime baltai, likusius žymime juodai. Kiek sunkiau įvertinti gradiento apibendrintus ekstremumus. Jei žymėtume tik tikrus gradiento ekstremumus, tai pažymėtų taškų vaizde atsirastų tik pavieniai pažymėti taškai ir negautumėme kontūrų. Todėl įvesime sąvoką apibendrinto ekstremumo.

1 apibrėžimas. *Fiksuokime spindulį $R > 0$ ir poziciją (x, y). Tarkime taškelio skritulinėje aplinkoje yra C diskrečių pozicijų nutolusių ne daugiau kaip per atstumą R. Fiksuokime diskretųjį vaizdą $u = u_{x,y}$ ir tegul c skritulinės aplinkos taškeliuose vaizdo reikšmės viršija vaizdo reikšmę centriniame taške. Tuomet sakysime, kad (x, y) yra diskretaus vaizdo $u_{x,y}$ c-osios eilės apibendrinto maksimumo taškas, jei $c <= C/2$. Jei $c > C/2.$, tokį taškelį Vadinsime $(C - 1 - c)$-osios eilės apibendrintas minimumo tašku.*

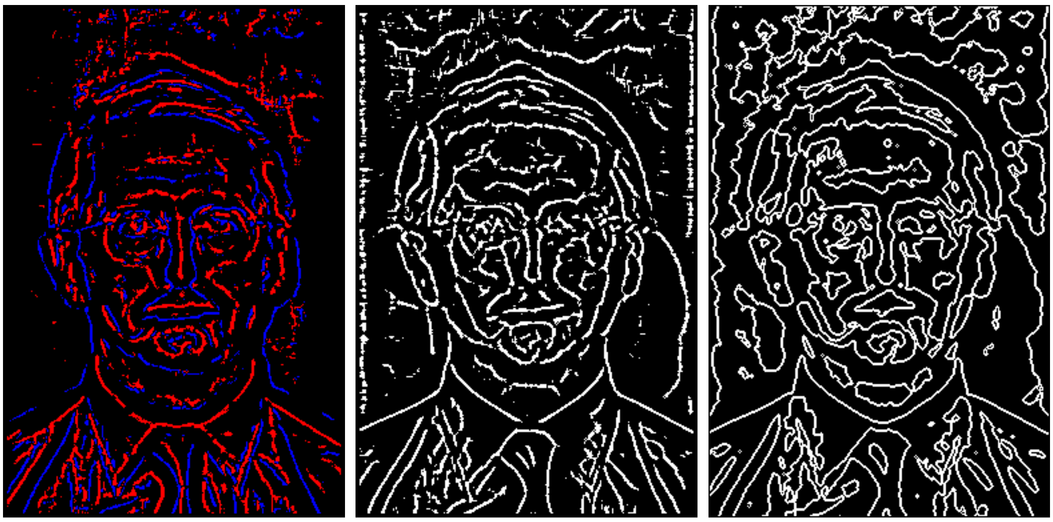
Pagal pateiktą apibrėžimą kiekvienas taškelis yra apibendrintas kokios nors eilės maksimumo arba minimumo taškeliu. Apibrėžiant kontūrą mes paliksime tik ekstremumų taškelius, kurių eilė yra nedidesnė už pasirinktą parametrą. Rekomenduojama fiksuoti $R = 3$ ir pažymėti visus apibendrintus ekstremumus, kurių eilė nedidesnį už 10. Tačiau jūs esate laisvi eksperimentuoti ir pasirinkti savo parametrus. Apibendrintus maksimumo taškelius žymėsime raudonai, o minimumo taškelius mėlynai. Tokiu būdu gausime dviejų spalvų kontūrų vaizdelį. 5 paveikslėlis iliustruoja raineles vidurkinto ir gradiento vaizduose išskirtus kontūrus, kurie sudaryti iš apibendrintųjų ekstremumų. Vidurkintame vaizde raudonai nuspalvinti apibendrinti lokalaus maksimumo taškai, o mėlynai minimumo taškai. Visais atvejais išskiriant reikšmingus apibendrintus lokaliuosius ekstremumus ėmėme $R = 3$ ir lokalius ekstremumus, kurių eilė nedidesnė kaip 10 ($c <= 10$). Taip pat pateiktas dar vienas kontūrų paveikslėlis gautas pažymint baltai Laplaso nulio kirtimo taškus.

1.4.1 Skaitmeninio vaizdo kontūrų išskyrimo praktinė užduotis

Parašyti programą, kuri apskaičiuotų duoto skaitmeninio vaizdo kontūrus. Pirma suglodinkite vaizdą eksponentiniu simetriniu filtru ir raskite jo gradiento modulio ir Laplaso vaizdus (pereita praktinė užduotis). Toliau vidurkintame išskirkite apibendrintus ekstremumus, gradientiniame vaizde apibendrintus lokalius maksimumus,



5 paveikslėlis: Vidurkintos rainelės apibendrinti lokalūs ekstremumai (kairėje viršuje), jos gradiento apibendrinti lokalūs maksimumai (dešinėje viršuje), Laplaso nulio kirtimai (apačioje)



6 paveikslėlis: Vidurkinto veido apibendrinti lokalūs ekstremumai (kairėje), jo gradiento apibendrinti lokalūs maksimumai (viduryje), Laplaso nulio kirtimai (dešinėje)

7 paveikslėlis: Vidurkinto piršto apibendrinti lokalūs ekstremumai (kairėje), jo gradiento apibendrinti lokalūs maksimumai (viduryje), Laplaso nulio kirtimai (dešinėje)

o Laplaso vaizde nulio kirtimus. Glodinimo parametrą σ, aplinkos spindulį R ir ekstremumų eilę C pasirinkite laisvai optimizuodami gautų kontūrų estetinį vaizdą.

Išskiriant kontūrus eksperimentuokite su pereitos užduoties vaizdais. Pasistenkite gauti vaizdus panašius į 5-4 pateiktuose paveikslėliuose gautus kontūrus.

Vertinama iki 0.75 egzamino balų. Jei dėl nepateisinamų priežasčių atsiskaitoma vėliau nei devynios savaitės nuo pirmųjų pratybų, už kiekvienos savaitės vėlavimą vertinimas mažinamas 0.25 balo.

1.5 Vaizdų požymių mažinimas ir papildomi atributai

Gautus skaitmeninio vaizdo kontūrų taškus laikysime jo požymiais. Išskiriant požymius siekiama mažinti jų kiekį atmetant mažai informatyvius ir pridedant atributus, didinančius požymių informatyvumą. Intuityviai aišku, kad gautus kontūrus, sudarytus iš požymių taškų, galima suploninti iki vieno taškelio storio praktiškai nesumažinant kontūrų taškais perteikiamos informacijos. Storų kontūrų, ar plotų ploninimas iki vieno taškelio storio jungių takų linijų vadinamas *skeletizavimu* [2]. Šiame kurse nesigilinsime ir nesieksime realizuoti kokio nors skeletizavimo algoritmo, o pasinaudosime jūsų kolegų skeletizavimo algoritmų kokia nors realizacija [3]. Turint skeletizuotą vaizdą, galima nutrinti labai trumpus jungius kontūrus, išskirti kontūrų ypatingus taškus, kurie charakterizuoja linijų išsidėstymą ir jų jungimosi topologiją. Paprasčiausias būdas išskirti ypatingus plonų linijų taškus yra fiksuoti linijos taškelį ir ieškoti kiek kartų jo aštuonių kaimynų aplinkoje pereinama iš baltos

	1	
1	*0*	1
	1	

	1		1
	0	1	
	1	1	
1			1

	1		
	0	*1*	1
	1	*0*	
1		1	

		1	
1	*0*	*1*	
	1	*0*	1
	1		

1			
	1		
	0	*1*	1
1	*1*	1	
			1

	1		1
	0	1	
	1	*0*	
1	*0*	*1*	1
			1

1 lentelė: Nepatogių ypatingų taškų klasifikacijai atvejų pavyzdžiai

į juodą arba atvirkščiai. Suradus šį skaičių, jis dalinamas iš dviejų ir gauta vertė K charakterizuoja linijos tašką. Šis metodas kartais vadinamas straipsnio autoriaus, kuris pirmasis jį paskelbė, Rutovič vardu [4]. Žemiau pateiktoje lentelėje nurodytos galimos K reikšmės ir jos interpretacijos.

K	**Taško interpretacija**
0	Pavienis
1	Pabaiga
2	Jungumo (neypatingas)
3	Išsišakojimo
4	Susikryžiavimo

Matematiškai tai galima užrašyti taip:

u_6	u_7	u_8
u_5	u_0	u_1
u_4	u_3	u_2

$K = \frac{1}{2}\sum_{i=1}^{8} |u_i - u_{i+1}|$, kur $u_9 = u_1$ ir $|u_i - u_{i+1}| = 0$, kai reikšmės sutampa, ir $|u_i - u_{i+1}| = 1$ priešingu atveju. Deja šis paprastas ypatingų taškų klasifikacijos būdas kartais duoda nenatūralius rezultatus.]reftab:blogiAtvejai lentelėje pateikta keletas pavyzdžių. Pasvirę 0 ir 1 taškeliai žymi pozicijas, kuriose gaunami nenatūralūs ypatingieji taškai. Tokių taškų eliminavimui taikomi papildomi vaizdų apdorojimo metodai [5].

Paliekant tolimesniam atpažinimui naudoti tik išskirtus ypatinguosius taškus ženkliai sumažiname požymių kiekį. Taip labai dažnai daroma pirštų atspaudų vaizdų palyginime. Tačiau net ir pirštų atspaudų atveju nepakanka vien tik informacijos apie ypatingojo taško tipą ir jo koordinates. Gana dažnai naudojami papildomi kontūro taškų atributai, kurie pagerina atpažinimo kokybę. Tarkime kontūras yra sudarytas iš gradiento modulio apibendrintųjų maksimumo taškų. Tokiuose taškuose gana patikimai įvertinama gradiento kryptis ir jos kvantuotą vertę galima laikyti papildomu požymio atributu. 8-10 paveikslėliai iliustruoja suplonintus gradientų

8 paveikslėlis: Rainelės gradiento modulio požymiai su krypties atributu ($\sigma = 2$)

kontūrus, kurie sudaryti iš apibendrintųjų maksimumų. Gaunant šiuos vaizdus, vidurkinant buvo naudoti (34) ir (35) formulėmis apskaičiuojami koeficientai a_0 ir b_0, o σ vertės buvo parenkamos individualiai kiekvienam paveikslėlio tipui. Apibendrintųjų maksimumų kontūrai buvo suploninti panaudojant skeletizavimo algoritmą [6]. Suplonintų kontūrų taškai nuspalvinti remiantis gradiento kryptimi. Gradiento vektorius yra sudarytas iš vidurkinto vaizdo išvestinių x ir y kryptimis. Šias išvestines anksčiau naudojome apskaičiuojant gradiento modulio(37) reikšmes. Spalvos kvantuojamos aštuoniomis lygiomis dalimis iš galimų gradiento vektoriaus polinio kampo reikšmių. Spalvų tvarka pagal laikrodžio rodyklę: RED, GREEN, BLUE, WHITE, YELLOW, CYAN, MAGENTA, GREY; pirma raudona spalva žymi kryptis, kurios sudaro su horizonto kryptimi kampą α nuo $-\pi/8$ iki $\pi/8$. Gradiento kampas su horizonto ašimi surandamas pagal formulę

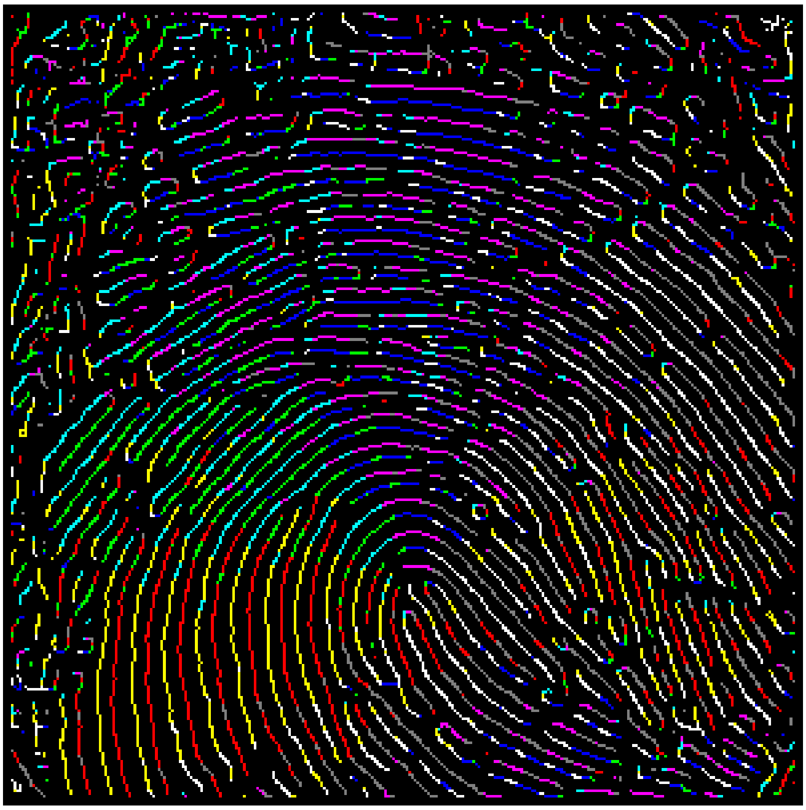
$$\alpha = \operatorname{atan2}(\overline{u}_y(x, y), \overline{u}_x(x, y)).$$

9 paveikslėlis: Veido gradiento modulio požymiai su krypties atributu ($\sigma = 3$)

Čia atan2 $=$ atan2(y, x) dviejų argumentų funkcija, kuri taško Dekarto koordinatėms (x,y) grąžina polinio kampo reikšmę. Ši funkcija yra realizuota visose programavimo kalbose; ja naudojantis atkreipkite dėmesį, kad kreipiantis reikia sukeisti Dekarto koordinačių tvarką.

1.5.1 Rainelių, veidų ir pirštų atspaudų vieningų požymių išskyrimo praktinė užduotis

Parašyti programą, kuri apskaičiuotų duoto skaitmeninio vaizdo suplonintus apibendrintų gradiento modulio maksimumų kontūrus. Kontūrus nuspalvinkite pagal kvantuotas aštuonias gradiento kryptis. Skaičiavimas naudokite ankstesnės užduoties programą. Vidurkinimo parametrą σ *parinkite savo nuožiūra individualiai kiekvieno tipo vaizdui, kad gautųsi estetiškesnis galutinių požymių vaizdas.*



10 paveikslėlis: Piršto atspaudo požymiai su krypties atributu ($\sigma = 1$)

Eksperimentuokite su pereitos užduoties vaizdais. Pasistenkite gauti vaizdus panašius į 8-10 paveikslėliuose gautus rainelės, veido ir piršto atspaudo požymių vaizdus.

Vertinama iki 0.75 egzamino balų. Jei dėl nepateisinamų priežasčių atsiskaitoma vėliau nei devynios savaitės nuo pirmųjų pratybų, už kiekvienos savaitės vėlavimą vertinimas mažinamas 0.25 balo.

Literatūra

[1] Bill Green, http://www.pages.drexel.edu/ weg22/edge.html

[2] Kálmán Palágyi, Vengrija, http://www.inf.u-szeged.hu/ palagyi/skel/skel.html

[3] K. Stukas, J. Janauskas, Š. Gruodis, M. Brašiškis, http://www.mif.vu.lt/ bastys/academic/ATE/skaiciai/skaiciu_atp.htm#Praktinis

[4] D. Rutovitz, Pattern Recognition, J. Roy. Statist. Soc., vol. 129, pp. 504-530, 1966.

[5] Feng Zhao and Xiaoou Tang, CISST02 International Conference, http://mmlab.ie.cuhk.edu.hk/2002/CISST02_Fingerprint.pdf, (Lokali kopija http://mif.vu.lt/ bastys/academic/ATE/pirshtai/CISST02_Fingerprint.pdf)

[6] T. Y. Zhang, C. Y. Suen, A fast parallel algorithm for thinning digital patterns, Communications of the ACM, v.27 n.3, p.236-239, March 1984.
Realizacija Java kalba: http://www.mif.vu.lt/atpazinimas/skaiciai/skelet/ZhangSuen.java