

## PRAEINANČIŲ ŽMONIŲ SKAIČIAVIMAS ĮTERPTINĖJE SISTEMOJE NAUDOJANT VAIZDO KAMERĄ

Nikolaj Kovaliov<sup>1</sup>, Andrius Ušinskas<sup>2</sup>

Vilniaus Gedimino technikos universitetas

El. paštas: <sup>1</sup>[nikolaj.kovaliov@gmail.com](mailto:nikolaj.kovaliov@gmail.com); <sup>2</sup>[andrius.usinskas@vgtu.lt](mailto:andrius.usinskas@vgtu.lt)

**Santrauka.** Vaizdų apdorojimas – daug išteklių ir laiko reikalaujantis uždavinys. Šiame straipsnyje optimizuojamos vaizdo apdorojimo funkcijos, nes įterptinės sistemos, kuri atliks šią užduotį, išteklių yra itin riboti. Nagrinėjamas algoritmas yra skirtas judantiems (išeinantiems ir einantiems) objektams skaičiuoti. Algoritmas įgyvendintas naudojant STM3210E-EVAL maketą ir CMOS kamerą OV7725. Gautas patenkinamas rezultatas – vaizdo apdorojimo sparta 0,5 kadro/s – gali būti gerinamas įvairius aplinkos apšvietimo kitimą.

**Reikšminiai žodžiai:** segmentavimas, vaizdų apdorojimas, judančių objektų skaičiavimas, įterptinė sistema, ARM mikrovaldiklis, CMOS kamera.

### Įvadas

Įterptinė sistema yra kompiuterinė sistema, skirta nustatyti funkcijai atlikti. Tokia sistema dažnai naudojama skaičiavimams vykdyti realiuoju laiku. Sistema skirta energiją taupantiems įrenginiams. Įterptinės sistemos yra daug pigesnės, energijos sąnaudomis priimtinesnės už *x86* architektūrą, tačiau atsilieka skaičiavimo sparta. Dėl minėtų įterptinės sistemos pranašumų plačiai taikomas vaizdų apdorojimo uždavinys – praeinančių žmonių skaičiavimas – įgyvendintas įterptinėje sistemoje.

Tokiam uždaviniui spręsti (Hall *et al.* 2005; Parks 2007) siūlo taikyti paprasčiausią fono išskyrimo metodą, kuriuo lyginamos fono skaisčio vertės su kadru, gautu iš vaizdo kameros RGB (angl. *Red-Green-Blue*) spalvų modelyje. Vėliau, panaudojus išdėtinimo ir erozijos operacijas, gaunami skaičiuojami objektai.

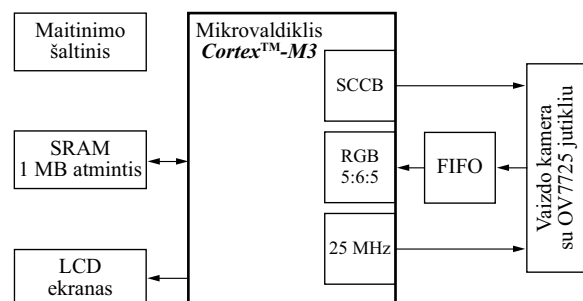
Haritaoglu *et al.* (2000) siūlo taikyti vadinamąjį *W4* metodą, kuriuo metu spalvinis fono modelis atvaizduojamas trimis vertėmis – maksimaliu intensyvumu, minimaliu intensyvumu ir maksimaliu intensyvumu tarp kadru. Toliau, taikant slenkstines funkcijas, gaunamas segmentuotas vaizdas, tuomet, taikant morfologinį filtravimą, gaunami suskaičiuojami objektai. Šio metodo privalumai yra paprastumas, o įgyvendintas algoritmas reikalauja mažai procesoriaus pajėgumų.

Wren *et al.* (1997) siūlo taikyti specialų Gauso (angl. *Gaussian*) modelį. Šis metodas pagrįstas tuo, kad kiekvienas taškas sudarytas iš Gauso skirstinio atsitiktinio skaičiaus.

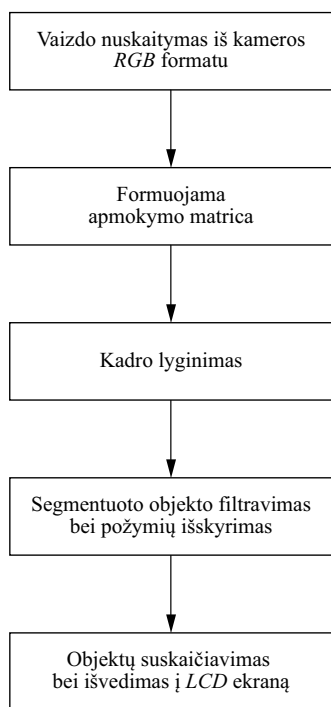
### Vaizdo apdorojimas

Darbe naudojamas STM3210E-EVAL maketas, turintis išorinę 1 MB (512×16 bitų) SRAM (angl. *Static Random Access Memory*) atmintį, kurioje bus saugoma iš kameros gauta informacija (ISSI... datasheet 2009; STM3210E... manual 2010) bei LCD (angl. *Liquid-crystal display*) skystųjų kristalų ekraną, kuriame bus vaizdinami duomenys.

Prie įterptinės sistemos (1 pav.) jungiama kamera OV7725, kuri konfigūruojama per SCCB (angl. *Serial Camera Control Bus*) sąsają (OMNIVISION... datasheet 2005). Iš kameros vaizdas nuskaitomas RGB565 formatu. Raudonai ir mėlynai spalvai skiriami po 5 bitus, o žaliai spalvai skiriami 6 bitai. Tokia spalvinė informacija saugoma dviem duomenų baitais. Vaizdo raiška yra QVGA (angl. *Quarter-VGA*), tokia kaip ir skystųjų kristalų ekranas. Vaizdo masyvas sudaro 320×240 vaizdo taškų ir užima 153 600 baitų SRAM atmintyje.



1 pav. Įterptinės sistemos struktūrinė schema  
Fig. 1. Structure of the embedded system



2 pav. Vaizdo apdorojimo sistemos struktūra

Fig. 2. Structure of the image processing based system

Vaizdo apdorojimo sistemos struktūra parodyta 2 pav. Iš kameros gauto vaizdo formuojama apmokymo matrica, kuri paskui naudojama kadrai palyginti. Tuomet atlikus segmentavimo, filtravimo ir požymių išskyrimo operacijas, gauti duomenys vaizdinami ekrane.

### Fono apmokymas

Fono apmokymas (3 pav.) vykdomas iš dešimties kadro, skirtų fonui apmokyti. Vaizdas iš RGB spalvų paletės į pustonių terpę paverčiamas pagal formulę:

$$G_R(\varphi) = R(\varphi) \times 0,2989 + G(\varphi) \times 0,5870 + B(\varphi) \times 0,1140, \quad (1)$$

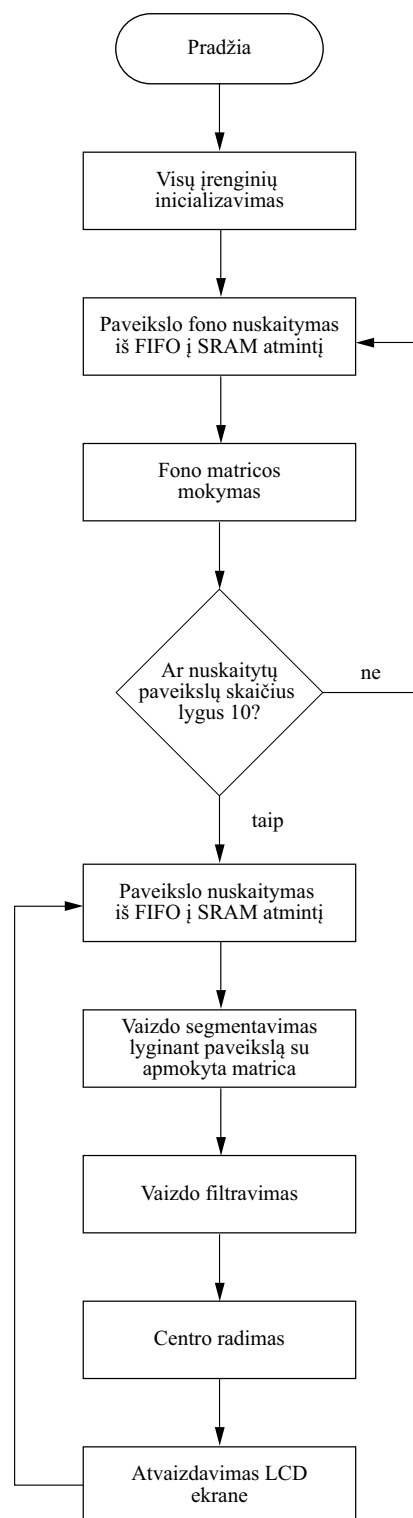
čia  $G_R(\varphi)$  – pustonių taško spalvos spalvinis intensyvumas;  $R(\varphi)$ ,  $G(\varphi)$ ,  $B(\varphi)$  – atitinkamai raudonos, žalios ir mėlynos spalvos taško spalvinis intensyvumas;  $\varphi$  – taško koordinatė.

### Vaizdo segmentavimas

Segmentuotas vaizdas gaunamas lyginant iš kameros gautą vaizdo matricą su fonu apmokyta vaizdo matrica, kuri susideda iš maksimalaus ir minimalaus intensyvumo. Kiekvienas taškas segmentuojamas kaip objektas arba kaip fonas naudojant lygtį:

$$F(\varphi) = (G_{R \max}(\varphi) > I(\varphi)) \& \& (G_{R \min}(\varphi) < I(\varphi)), \quad (2)$$

čia  $F(\varphi)$  – objektui segmentuotas taškas;  $G_{R \min}(\varphi)$  ir  $G_{R \max}(\varphi)$  – minimalus ir maksimalus intensyvumas.



3 pav. Praeinančių žmonių skaičiavimo įterptinės sistemos programos algoritmas

Fig. 3. Algorithm of the embedded system program that counts passing people

Segmentuotame vaizde atsiranda triukšmo. Atsiradęs triukšmas šalinamas vaizdo filtru.

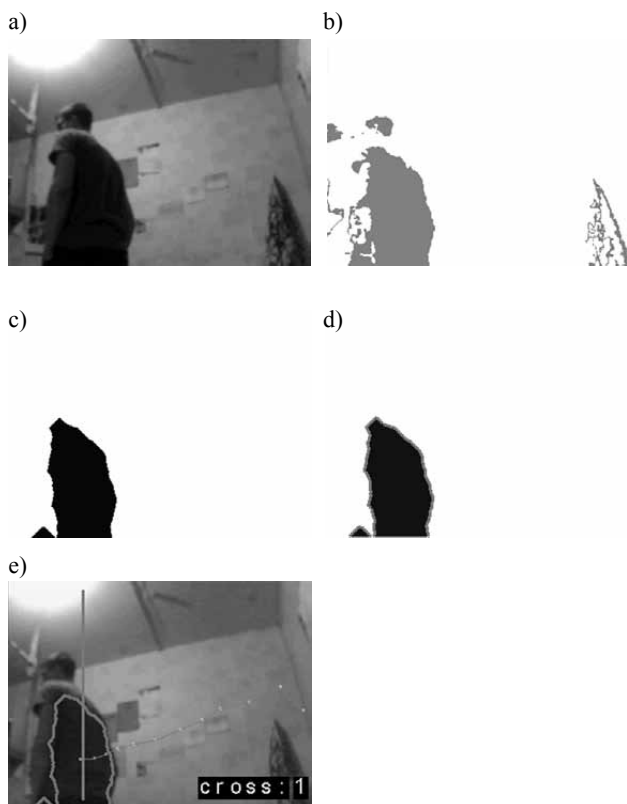
## Vaizdo filtravimas

Haritaoglu *et al.* (2000) vaizdui filtruoti siūlo naudoti morfologines funkcijas, tokias kaip ęsdinimą (angl. *erosion*) ir priauginimą (angl. *dilation*). Tačiau toks filtravimas yra imlus procesoriaus ištekliams. Dėl šios priežasties bus filtruojami gretutiniai taškai, tuo metu tiesiogiai tikrinami centrinio taško ir gretimų taškų vertės, taip išfiltruojami vienetiniai taškai ir smulkios linijos. Šis filtravimas mažiau apkraus procesorių dėl trumpesnės kreipties į atmintį. Gretutinių taškų filtravimas gali dirbti 20 kartų greičiau nei taikant morfologines funkcijas.

## Objekto centro nustatymas

Objekto centrui nustatyti taikomas aštuonių krypčių rekursinis užpildymo algoritmas (angl. *flood fill*), šiuo atveju yra sumuojamos visos objekto taškų koordinatės, atskirai  $x$  ašiai bei  $y$  ašiai (Lode's... 2004). Koordinatžių suma dalijama iš objekto taškų skaičiaus, gautas rezultatas yra objekto centro koordinatės.

Gautos objekto koordinatės saugomos atskirame masyve. Taip pat saugomos kitų kadro, centrų koordinatės tame pačiame masyve. Masyvas išvalomas, kai objekto centras patenka už kadro ribų. Objekto judėjimui atvaiz-



4 pav. Vieno kadro apdorojimo vaizdai: a – pradinis; b – po segmentavimo; c – filtruotas; d – po kraštų radimo; e – fiksuojantis judantį objektą

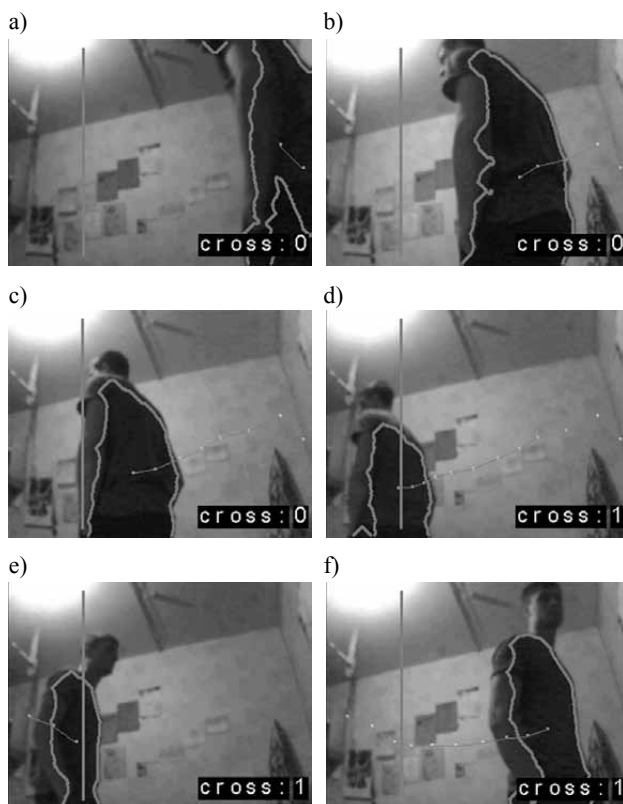
Fig. 4. Images of one processed frame: a – captured; b – segmented; c – filtered; d – after edge detection; e – moving object counted

duoti panaudota tiesė. Tiesei bręžti panaudotas Bresenhamo tiesės algoritmas (Hexar 2007). Objekto kontūrai nubręžti skaičiuojami objekto krašto taškai.

## Rezultatai

4 pav. pateikti rezultatai iliustruoja vieno kadro nuskaitymą, jo segmentavimą, filtravimą ir centro koordinatžių skaičiavimą. Triukšmas, matomas 4 pav. b dalyje, atsirado dėl kintančio apšvietimo ar klaidingo fono apmokymo. 4 pav. c dalyje matyti, kad triukšmas yra pašalintas. Filtruojant triukšmą pašalinama viršutinė judančio objekto dalis. Ji nufiltruojama dėl per mažo dydžio. Visi objektai, turintys mažiau nei 100 taškų, yra pašalinami. 4 pav., d, parodyti rasti objekto kraštai ir nustatytas objekto centras. Pagal centro koordinatės algoritmas ieško, ar objektas peržengė nustatytą kirtimo liniją. 4 pav., e, parodytas objektas, peržengęs kirtimo liniją, todėl skaitiklis *cross* padidintas (iš 0 į 1).

5 pav. pateikti judančio objekto sekimo rezultatai. Objektas, judantis iš dešinės į kairę bei kertantis nustatytą liniją, yra sumuojamas prie kirtusių liniją judančių objektų skaičiaus (4 pav., d). Tačiau judantis objektas, kertantis nustatytą liniją iš kairės į dešinę, nėra sumuojamas (4 pav., f).



5 pav. Objekto sekimas: a–d – judama iš dešinės į kairę, todėl objektas registruojamas; e–f – judama priešinga kryptimi

Fig. 5. Object tracking: a–d – movement from right to left thus object is registered; e–f – movement in the opposite direction

## Išvados

1. Nenaudojant morfologinių filtrų ar kitų stipriai procesoriaus darbą apkraunančių algoritmų, pateikiamas būdas skaičiuoti įeinančius ar išeinančius žmones vaizde taikant CMOS OV7725 vaizdo kamerą. Įvertinus procesoriaus stm32f103zet6 nedidelį 72 MHz taktinį dažnį gaunamas 0,5 kadro per sekundę greitis, kurį galima laikyti patenkinamu tokio galingumo įterptinei sistemai. Straipsnyje aprašytas metodas aptinka visus įeinančius ar išeinančius žmones, apskaičiuoja jų vaizdų centrus, suskaičiuoja įeinančius ar išeinančius žmones, kertančius nustatytą liniją.
2. Algoritmo apdorojimo sparta labiausiai didėja keičiant morfologinę ęsdinimo funkciją paprastu gretutinių taškų filtru.
3. Algoritmą galima tobulinti jį papildžius lėto apšvietimo pokyčių adaptacija, kurios metu fono matrica apmokoma periodiškai, tačiau tai papildomai apkraus procesorių.

## Literatūra

- Hall, D.; Nascimento, J.; Ribeiro, P.; Andrade, E.; Moreno, P.; Pesnel, S.; List, T.; Emonet, R.; Fisher, R. B.; Santos-Victor, J.; Crowley, J. L. 2005. Comparison of target detection algorithms using adaptive background models, in *2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 15–16 October, 2005, 113–120. <http://dx.doi.org/10.1109/VSPETS.2005.1570905>
- Haritaoglu, I.; Harwood, D.; Davis, L. S. 2000. Real-time surveillance of people and their activities, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (8): 809–830.
- Hexar. 2007. *Drawing Lines – The Bresenham Algorithm*. *Graphics Tutorial* [interaktyvus], [žiūrėta 2013 m. balandžio 21 d.]. Prieiga per internetą: [http://www.cs.toronto.edu/~smalik/418/tutorial2\\_bresenham.pdf](http://www.cs.toronto.edu/~smalik/418/tutorial2_bresenham.pdf)
- ISSI IS61WV51216BLL 512K × 16 high-speed asynchronous CMOS SRAM datasheet [interaktyvus]. 2009 [žiūrėta 2013 m. balandžio 21 d.]. Prieiga per internetą: <http://www.mouser.com/ds/2/198/61-64WV51216-24281.pdf>
- Lode's Computer Graphics Tutorial. *Flood fill algorithm* [interaktyvus]. 2004 [žiūrėta 2013 m. balandžio 21 d.]. Prieiga per internetą: <http://lodev.org/cgtutor/floodfill.html>
- OMNIVISION OV7725 Color CMOS VGA (640×480) CAMERACHIP™ Sensor datasheet [interaktyvus]. 2005 [žiūrėta 2013 m. balandžio 21 d.]. Prieiga per internetą: [https://www.zhopper.narod.ru/mobile/ov7725\\_full.pdf](https://www.zhopper.narod.ru/mobile/ov7725_full.pdf)
- Parks, D. 2007. *EECE575: Project Proposal: Effects of Post-processing on Background Subtraction Algorithms*, 1–3.
- STM3210E-EVAL evaluation board manual [interaktyvus]. 2010 [žiūrėta 2013 m. balandžio 21 d.]. Prieiga per internetą: [http://www.st.com/st-web-ui/static/active/jp/resource/technical/document/user\\_manual/CD00178166.pdf](http://www.st.com/st-web-ui/static/active/jp/resource/technical/document/user_manual/CD00178166.pdf)
- Wren, C. R.; Azarbayejani, A.; Darrell, T.; Pentland, A. P. 1997. Pfindex: real-time tracking of the human body, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(7): 780–785. <http://dx.doi.org/10.1109/34.598236>

## COUNTING OF MOVING OBJECTS USING EMBEDDED SYSTEM AND CMOS CAMERA

N. Kovaliov, A. Ušinskas

Abstract

Image processing is inescapable and long process required for image recognition. To implement it in an embedded system with limited resources, process optimizations should be made. This paper presents an algorithm for counting incoming and outgoing moving objects. The algorithm is implemented in STM3210E-EVAL development board, where image processing is done.

**Keywords:** segmentation, image processing, counting of moving objects, embedded system, ARM microcontroller, CMOS camera.