

Eiropas Reģionālās attīstības fonds

Prioritāte: 2.1. Zinātne un inovācijas

Pasākums: 2.1.1.1. Zinātne, pētniecība un attīstība

Aktivitāte: 2.1.1.1. Atbalsts zinātnei un pētniecībai

Projekta nosaukums: „Multimodālas biometrijas tehnoloģija drošai un ērtai personu
autentifikācijai” (BiTe)

Līguma noslēgšanas datums: 10.12.2010.g.

Projekta sākuma datums: 01.01.2011.g.

Projekta beigu datums: 31.12.2013.g.

Vienošanās Nr.2010/0285/2DP/2.1.1.1.0/APIA/VIAA/098

Eiropas Reģionālās attīstības fonda finansējuma saņēmējs: Elektronikas un
datorzinātņu institūts (EDI)

ZINĀTNISKĀ PĒTĪJUMA PROGRESU APLIECINOŠĀ DOKUMENTĀCIJA

Pārskata numurs Nr.5. par periodu no 01.03.2012.g. līdz 30.06.2012.g.

Projekta zinātniskais vadītājs: Modris Greitāns, Dr.sc.comp., vad. pētnieks
Pētījuma projekta izpildītāju saraksts: Mihails Broitmans, Dr.sc.comp., vad. pētnieks
Rihards Fuksis, pētnieks
Mihails Pudžs, pētnieks
Rinalds Ruskuls, asistents
Oļegs Ņikišins, asistents
Zanda Seržāne, asistente
Artūrs Kadiķis, programmēšanas inženieris
Dāvis Barkāns, elektronikas inženieris
Teodors Eglītis, informātikas/elektronikas tehniķis

Satura rādītājs

Anotācija	3
Ievads	3
Rezultātu kopsavilkums.....	3
Aktivitāte: Lietišķie pētījumi	5
Sejas un plaukstu attēlu iegūšanas metožu izpēte.....	5
Biometrisku datu apstrādes algoritmu darbības izvērtēšana.....	9
Biometrisku datu iegūšanas algoritmu paralelizācija un implementēšana programmējamās loģiskās masīvos	9
Biometrisku datu kriptēšanas, glabāšanas un lietojuma apakšsistēmas izveide	12
Atsevišķu sistēmas komponentu un bloku izstrāde un izpēte	18
Aktivitāte: Eksperimentālā izstrāde.....	20
Multimodālas biometrijas tehnoloģijas koncepta definēšana	20
Tehnoloģijas demonstratora eksperimentālā maketa montēšana.....	21
Algoritmu implementēšana eksperimentālajā maketā	22
Secinājumi	23
Pielikumi.....	25

Anotācija

BiTe ir ERAF līdzfinansēts projekts zinātnei un pētniecībai. Projekta mērķis ir droša, ērta un plaši pielietojama personu identifikācijas risinājuma izveide. Projekts ietver sevī sekojošas aktivitātes: Lietišķos pētījumus biometrijas parametru ieguvē, apstrādes metožu implementēšanā ierobežotu resursu sistēmās, atsevišķu sistēmas komponentu un bloku izstrādē; Eksperimentālos pētījumus tehnoloģiju demonstratora izstrādē; Rūpniecisko tiesību aizsargāšana, patenta pieteikums; Projekta publicitāte. Šajā dokumentā dots pārskats par projekta piektajā periodā (01.03.2012.- 30.06.2012) veiktajiem pētniecības darbiem un šobrīd sasniegtiem rezultātiem.

Projektu atbalsta Eiropas Reģionālās attīstības fonds, līguma Nr. 2010/0285/2DP/2.1.1.1.0/10/APIA/VIAA/098

Ievads

Identitātes nozagšana ir viens no modernajiem noziegumiem, kas „zaglim” dod iespēju radīt personām finansiālus zaudējumus. Katru gadu identitātes nozagšana pasaulē rada aptuveni 221 miljardu dolāru zaudējumus. Lielu interesi par biometrijas izmantošanu personu identifikācijā izrāda banku sektors. Bankas atzīst, ka šādas sistēmas ieviešana dotu iespēju aizstāt (vai kombinēt) uz parakstu vai PIN kodu ievadi balstītu norēķinu karšu autorizāciju ar kartes īpašnieka biometrisku atpazīšanu. Radot ērtu, lētu un drošu personu identifikācijas risinājumu, ir plašas iespējas to ieviest ikdienas dzīvē – bankās, tirdzniecības vietās, objektu piekļuves un lietošanas kontrolē (ieskaitot autotransportu) u.c.

BiTe pētniecības grupas darbs ir saistīts ar augstāk minēto problēmu risinājumu. Šajā projekta pārskata posmā ir veikti darbi un sasniegti rezultāti sekojošos aktivitātes „Lietišķie (rūpnieciskie) pētījumi” pētījumos:

- Sejas un plaukstas attēlu iegūšanas metožu izpēte;
- Biometrisku datu apstrādes algoritmu darbības izvērtēšana;
- Biometrisku datu kriptēšanas, glabāšanas un lietojama apakšsistēmas izveide;
- Atsevišķu sistēmas komponentu un bloku izstrāde un izpēte.

No 1.04.2012. uzsākti darbi aktivitātes „Eksperimentālā izstrāde” ietvaros:

- Multimodālas biometrijas tehnoloģijas koncepta definēšana;
- Tehnoloģijas demonstratora eksperimentālā maketa montēšana;
- Algoritmu implementēšana eksperimentālajā maketā.

Rezultātu kopsavilkums

Projekta piektajā periodā (01.03.2012.-30.06.2012.) veikto pētniecības uzdevumu un rezultātu kopsavilkums ir sekojošs:

- Sejas un plaukstas attēlu iegūšanas metožu izpēte – iepriekšējos pārskata periodos tika izpētīts kā izveidot augstas jutības infrasarkanās un redzamās gaismas attēlu

iegūšanas sistēmu; apskatītas plaukstu specifisko pazīmju izdalīšanas iespējas; izpētīti plaukstu ģeometrijas izmēri, kas minimāli atkarīgi no plaukstu stāvokļa un uzsākts plaukstu ģeometrijas algoritmu apraksts MATLAB vidē. Šajā pārskata periodā tika analizēta mērķa principa metodika un veikta vienlaicīga oriģinālu un filtrētu bilžu iegūšana.

- Biometrisku datu apstrādes algoritmu darbības izvērtēšana – iepriekšējos periodos tika realizēta 1. versija 2D sejas atpazīšanas algoritmam (sejas detektēšana, lokalizācija un atpazīšana), kas implementēts iegultā sistēmā; tika veidots automātisks sejas atpazīšanas algoritms signālprocesoram TMS320C6416; tika pievienoti galvenie elementi kameras modulim un izveidotas programmas attēla vizualizācijai un kvalitātes novērtēšanai. Piektajā periodā padziļināti tika izpētīts sejas detektēšanas algoritms, darbam daudzkodolu procesorā un analizētas sejas atpazīšanas algoritmu implementācijas iespējas.
- Biometrisku datu iegūšanas algoritmu paralelizācija un implementēšana programmējamajos loģiskos masīvos – iepriekš tika izveidota uz FPGA bāzēta attēlu ieguves sistēma; izstrādāta iespiedplate sejas atpazīšanas algoritmu testēšanai; projektēts 2D filtrs (konvolūcijas veikšanai FPGA platformā); veikta attēla filtrēšana reālā laikā; izveidota jauna sistēmas (125MHz) arhitektūra un izpildīta jaunā Aptina MT9V024 attēlu sensora montāža. Šajā periodā tika apskatīta datora saskarne ar prototipu.
- Biometrisku datu kriptēšanas, glabāšanas un lietojuma apakšsistēmas izveide – iepriekšējos pārskata periodos tika izanalizēti algoritmi *biohash* funkcijas skaitļošanai; izveidots algoritms, kas iegūst vidēji tādu pašu EER kā salīdzinot nešifrētus datus; izstrādāta programmatūra, kas salīdzina nešifrētus un šifrētus datus uz Java viedkartes; izstrādāts protokols, kas ļauj izveidot drošu sakaru kanālu starp viedkarti un autentifikācijas iekārtu; sastādīts ROI noteikšanas algoritms un apskatīts Rīda–Solomona kļūdu koriģējošs algoritms. Šajā periodā Biohash algoritmam tika apskatīta papildus informācija, ko pievienot datu vektoram, lai varētu uzlabot atpazīšanu un izveidota viedkartes saskarne.
- Atsevišķu sistēmas komponentu un bloku izstrāde un izpēte – iepriekšējos periodos tika veidotas funkcionālo bloku elektriskās principiālās shēmas *Altium Designer* vidē; izveidota principiālā viedkaršu komunikācijas maketa shēma; uzsākta kameras moduļa iespiedplates 2. versijas montāža; tika programmēti Aptina MT9V024 attēlu sensora reģistri; projektēts biometrijas sistēmas ietvars un izveidota iespiedplate viedkaršu saskarnes nodrošināšanai ar FPGA izstrādes rīku. Šajā pārskata periodā tika programmēts mikrokontrolieris MSP430 un realizēti LBP, LDP algoritmi MATLAB vidē.

Šajā pārskata periodā tika uzsāka aktivitāte „Eksperimentālā izstrāde” un šīs aktivitātes ietvaros uzsākts darbs vairākās apakšaktivitātēs. To rezultātu kopsavilkums ir sekojošs:

- Multimodālas biometrijas tehnoloģijas koncepta definēšana – tika aprakstīts prototipa funkcionālo bloku (ALTERA DE-2 izstrādes rīks, mikrokontroliera bloks, optiskā filtra pārlēgšanas bloks, FTDI USB bloks, viedkaršu saskarnes bloks un displejs) darbības princips un savstarpējā sadarbība.
- Tehnoloģijas demonstratora eksperimentālā maketa montēšana – šajā pārskata periodā tika veikta biometrijas prototipa 1.versijas montāža, kur tika savienoti galvenie sistēmas bloki. Visi sistēmas bloki ievietoti organiskā stikla ietvarā, veidojot vienotu prototipēšanas sistēmu.
- Algoritmu implementēšana eksperimentālajā maketā – ir veiktas CMF modifikācijas; nomainīts maskas izmērs no 9x9 uz 15 x15; izmainīta pati maska un gūts priekšstats par VHDL parametrizējamām ķēdēm.

Turpmāk dokumentā ir detalizēti aprakstītas katras uzdevumu grupas pētniecības darbības un rezultāti.

Aktivitāte: Lietišķie pētījumi

Sejas un plaukstu attēlu iegūšanas metožu izpēte

Mērķa principa metodikas analīze

Atskaites periodā tika izanalizēta «mērķa principa» metodikas efektivitāte, kas piedāvāta plaukstu ģeometrisko raksturlielumu līdzības novērtēšanai. Analīzes rezultāti parādīja, ka šī raksturlielumu salīdzināšanas metodika uzrāda pieņemamus rezultātus tajos gadījumos, kad jebkurš izmērītais i -tais raksturlielums f_i atšķiras no k -tā etalona i -tā raksturlieluma vērtības θ_{ki} , ne vairāk, kā divas reizes. Pretējā gadījumā līdzības koeficienta nepārtrauktība tiek pārtraukta (kļūst bezgalīga):

$$D_{ki} = D_{ki}, \text{ ja } D_{ki} \leq P_i \text{ pretējā gadījumā } D_{ki} = \infty.$$

šeit P_i – koeficients, kas nosaka nepieciešamo precizitātes līmeni i -tā parametra salīdzināšanai.

Šīs nepilnības novēršanai un līdzības koeficienta izmaiņu nepārtrauktības nodrošināšanai, tiek piedāvāta jauna formula. Plaukstu pētāmo ģeometrisku raksturlielumu kopu, kas tiks izmantota identifikācijai, apzīmēsim ar $F = \{f_1, f_2, \dots, f_N\}$, kur f_i – i -tā raksturlieluma nomērītās vērtības. Bet k -tās personas vidējo raksturlielumu kopu no datubāzes: $\theta_k = \{\theta_{k1}, \theta_{k2}, \dots, \theta_{kN}\}$, θ_{ki} – k -tās personas i -tā raksturlieluma vidējā vērtība. Raksturlieluma i salīdzināšanai, ar k -tā etalona atbilstošo raksturlielumu, piedāvājam izmantot sekojošu formulu:

$$D_{ki} = 1 - \frac{|f_i - \theta_{ki}|}{f_i + \theta_{ki}}$$

kur D_{ki} – i -tā nomērītā plaukstu ģeometriskā raksturlieluma līdzības koeficients k -tās personas i -tā raksturlieluma vidējai vērtībai. Atšķirībā no iepriekšējās formulas, šajā formulā līdzības koeficients ir nepārtraukts visā reālo skaitļu apgabalā. Ja k -tās personas i -tā raksturlieluma nomērītā vērtība sakrīt ar i -tā raksturlieluma vidējo vērtību (no datubāzes), tad skaitītājs formulā būs vienāds ar nulli, bet līdzības koeficients $D_{ki} = 1$. Savukārt, ja $f_i \neq \theta_{ki}$, tad līdzības koeficients būs mazāks nekā viens ($D_{ki} < 1$). Pie tam, jo vairāk atšķirsies i -tā

raksturlieluma nomērītā vērtība no k -tās personas i -tā raksturlieluma vidējās etalonvērtības, jo mazāka būs līdzības koeficienta vērtība.

Turpinājumā nepieciešams noteikt, cik lielā mērā mērīto raksturlielumu kopums atbilst k -tās personas etalonraksturlielumu kopumam (no datubāzes). Šim nolūkam paredzēts izmantot sekojošu formulu:

$$D_k = D_{k1} * D_{k2} * \dots * D_{kN}$$

kur D_k – izmērīto raksturlielumu aprēķinātais līdzības koeficients k -tās personas raksturlielumiem no datubāzes. Ideālā gadījumā, kad mērījumi ir absolūti precīzi un nomērītie raksturlielumi ir k -tās personas raksturlielumi, līdzības koeficienta vērtība būs vienāda ar vienu ($D_k = 1$). Pretējā gadījumā $D_k < 1$.

Ņemot vērā, ka ģeometrisko raksturlielumu salīdzināšanu paredzams izmantot kā iepriekšēju identifikācijas etapu, pēc kura tiks veikta detalizētāka salīdzināšana, lietderīgi ievest vairākus nomērīto un etalonraksturlielumu atbilstības (līdzības) līmeņus. Atkarībā no nepieciešamās vērtību atbilstības precizitātes šie līmeņi var tikt attiecīgi iestādīti. Piedāvājam ievest sekojošus līdzības līmeņus:

- 1) SAKRIŠANA (match);
- 2) IESPĒJAMA SAKRIŠANA (possible match);
- 3) NESAKRIŠANA (unmatch).

Trīs, un ne vairāk, līmeņu izvēle pamatojama ar identifikācijas algoritma kopējo uzbūvi, kura ietvaros, ģeometrisko raksturlielumu salīdzināšana ir tikai iepriekšējs etaps, kura rezultāti izmantojami, lai tiktu noteikta nepieciešamība izpildīt tālākas pārbaudes. Tā, piemēram, ja noteikts līdzības līmenis - «SAKRIŠANA», tad var neveikt ROI precizētu asinsvadu analīzi. Ja noteikts līdzības līmenis «IESPĒJAMA SAKRIŠANA», tad nepieciešams pāriet pie nākošajām personas identifikācijas procedūrām. Visbeidzot, noteikts līdzības līmenis «NESAKRIŠANA», tad, iespējams, nav nozīmes veikt detalizētākas identifikācijas procedūras.

Apzīmēsim ievestos līdzības līmeņus sekojoši: «SAKRIŠANA» - S_m , «IESPĒJAMA SAKRIŠANA» - S_{pm} , «NESAKRIŠANA» - S_{um} .

Līdzības līmeņu noteikšanu var veikt sekojoši: līdzības līmenis S_m nozīmē, ka visi izmērītie ģeometriskie raksturlielumi atšķīrušies no etalona ne vairāk, kā par pieļaujamo mērījumu neprecizitāti. Pieņemsim, ka maksimāli pieļaujamā katra raksturlieluma mērījuma neprecizitāte sastāda R procentus. Tad:

$$S_m \subset \left\{ D_k^R \right\}, \quad \text{kur } D_k^R = D_{k1}^R * D_{k2}^R * \dots * D_{kN}^R$$

un

$$D_{ki}^R = 1 - \frac{|f_i^R - \theta_{ki}|}{f_i^R + \theta_{ki}}, \quad |f_i^R - \theta_{ki}| \leq \frac{R}{100}, \quad i \in \{1, 2, \dots, N\}$$

Kā piemēru apskatīsim gadījumu, kad mērījumu neprecizitāte sastāda ne vairāk kā 10% no atbilstošās etalona vērtības. Tas nozīmē, ka

$$|f_i^R - \theta_i| \leq 0.1 \quad \text{un} \quad 0.047 \leq \frac{|f_i^R - \theta_i|}{f_i^R + \theta_i} \leq 0.053$$

No tā izriet, ka

$$D_{ki}^{10''} \approx 0.95, \quad i = \overline{1, N} \quad \text{un} \quad D_k^{10''} = (0.95)^N$$

kur N - plaukstas izmērāmo un salīdzināmo ģeometrisko parametru skaits.

Analogā veidā var noteikt S_{pm} un līdzības koeficientu robežas:

$$S_{pm} \subset \left\{ D_k^T \right\}, \quad \text{kur} \quad D_k^T = D_{k1}^T * D_{k2}^T * \dots * D_{kN}^T$$

kur T – maksimāli pieļaujamā izmērītā raksturlieluma atšķirība no etalona, pie kura vēl varētu būt iespējams, ka izmērītais raksturlielums atbilst k-tās personas etalonraksturlielumam no datubāzes. Šajā gadījumā:

$$D_{ki}^T = 1 - \frac{|f_i^T - \theta_{ki}|}{f_i^T + \theta_{ki}}, \quad \frac{R}{100} < |f_i^T - \theta_{ki}| \leq \frac{T}{100}, \quad R < T, \quad i \in \{1, 2, \dots, N\}$$

Ja katra izmērītā raksturlieluma f_i vērtība (visiem i) atrodas diapazonā:

$$\left(\frac{\theta_{ki}(100-T)}{100} \leq f_i < \frac{\theta_{ki}(100-R)}{100} \right) \mid \left(\frac{\theta_{ki}(100+R)}{100} < f_i \leq \frac{\theta_{ki}(100+T)}{100} \right), \quad i \in \{1, 2, \dots, N\}$$

tad līdzības koeficients ietilpst apakškopā S_{pm} . Ja, piemēram, pieņem, ka T=25%, tad izmērītā raksturlieluma vērtību diapazons, pie kura tiek apmierināta iepriekšminētā nevienādība un sekojoši līdzības koeficients, kas ietilpst apakškopā S_{pm} , būs sekojošs:

$$(0.75 * \theta_{ki} \leq f_i < 0.90 * \theta_{ki}) \mid (1.1 * \theta_{ki} < f_i \leq 1.25 * \theta_{ki}), \quad i \in \{1, 2, \dots, N\}$$

Ja izpildās vismaz viena no sekojošajām nevienādībām:

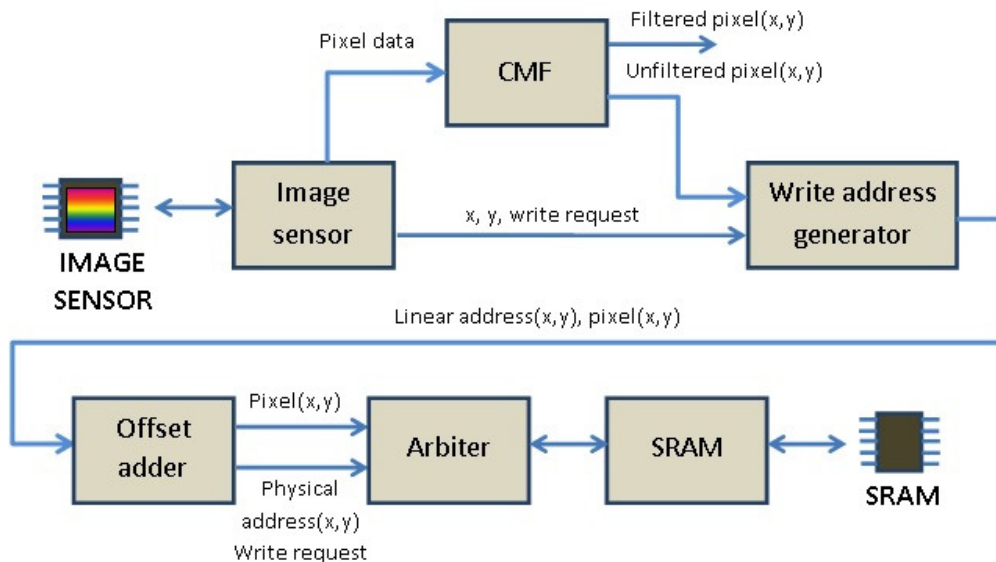
$$(f_i < 0.75 * \theta_{ki}) \mid (f_i > 1.25 * \theta_{ki}), \quad i \in \{1, 2, \dots, N\}$$

kaut vai vienam f_i ar pieņemtajām vērtībām R un T (R=10%, T=75%), tad līdzības koeficients nokļūst apakškopā S_{um} .

Kopējā plaukstu ģeometrisko raksturlielumu kopu identifikācijas algoritma blokshēma parādīta pielikumā Nr.1.

Oriģinālu (nefiltrētu) un filtrētu bilžu vienlaicīga iegūšana.

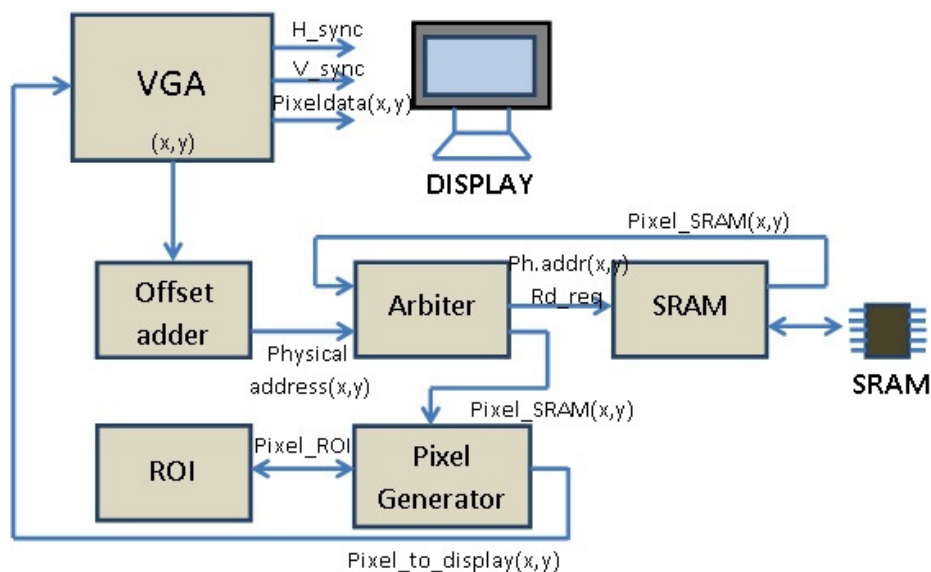
Līdz šim mūsu izveidotā arhitektūra nodrošināja viena veida bildes iegūšanu, tas ir, vai oriģinālu (nefiltrētu) vai filtrētu ar CMF. Apskatot sistēmas blokus, tika izsecināts, ka ir iespējams ierakstīt oriģinālu un filtrētu bildi SRAM integrālajā mikroshēmā. SRAM atmiņai ir 16 bitu plata datu kopne. Tas ir iespējams, jo gan oriģinālas, gan filtrētas bildes viena pikseļa platums ir 8 biti. Šāds bitu daudzums izvēlēts, jo lielāku bitu daudzumu nav iespējams attēlot uz izvēlētā izstrādes rīka – DE2-115.



1.attēls. Bildes iegūšanas blokshēma

Lai būtu iespējams izvēlēties starp bildes veidiem (vai oriģinālu vai filtrētu) bija jāmodificē attēlu sūtīšanas un attēlošanas modulis kā arī jāpapildina "Manager" modulis.

"Manager" modulis tika papildināts ar kontroles reģistriem, kuri nosaka kādu bildi attēlot uz ekrāna, bet kāda bilde tiks nosūtīta uz datoru.



2.attēls. "Manaher" moduļa blokshēma

Attiecīgi pirms attēlošanas moduļa tika ielikt multiplexors, kas sadalīja nolasīto datu plūsmu no SRAM pa 8 bitiēm. Apakšējie 8 biti nozīmē oriģinālais, augšējie filtrētais attēls.

Turpmāk plānots:

- izstrādāt MATLAB kodu ģeometrisku raksturlielumu kopu identifikācijas algoritma blokshēmai;
- ģeometrisku raksturlielumu algoritmu izpēti;
- attēlu iegūšanas metodes izstrāde.

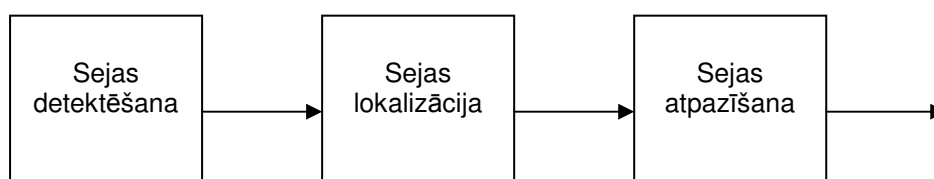
Biometrisku datu apstrādes algoritmu darbības izvērtēšana

Sejas atpazīšanas sistēmas realizācijai iepriekš tika izvēlēts TMS320C6416 signālprocesors, tomēr veicot padziļinātu procesora un algoritmu izpēti tika atklāta vajadzība pēc papildus datu apstrādes jaudas. Apsvērtas iespējas sistēmas realizācijai izmantot daudzkodolu signālprocesoru TMS320C6678 vai FPGA-balstītu sistēmu. Katrai no iepriekšminētajām sistēmām ir savas priekšrocības un trūkumi:

- FPGA ļauj panākt ātrdarbīgu paralēlu procesu izpildi, tomēr risinājums nav pietiekoši elastīgs un grūti modificējams.
- Texas Instrument signālprocesors TMS320C6678 satur 8 kodolus, kas arī ļauj veikt paralēlo apstrādi, bet tomēr ātrdarbība un paralelizēšanas iespējas nav tik plašas kā FPGA gadījumā.

Risinājums uz TMS320C6678 ir elastīgāks un ļauj veikt programmēšanu augstākā līmenī, kas prasītu mazāk programmēšanas resursus un tika pieņemts lēmums implementāciju balstīt signālprocesorā. Abi risinājumi ietver paralelizēšanas iespējas, tāpēc lielākais uzsvars šajā atskaitēs periodā tika likts uz algoritmu paralelizāciju.

Automātiskais sejas atpazīšanas algoritms sastāv no trim stadijām: sejas detektēšana, sejas lokalizācija un sejas atpazīšana:



Sejas detektēšanas un lokalizācijas uzdevumi balstās uz LBP un neironu tīklu kombinācijas (*Artificial Neural Network* - ANN). Sejas atpazīšanas algoritms balstās uz LBP un *Weighted Nearest Neighbour Classifier* (WNNC). LBP histogrammu rēķināšana ir sejas detektēšanas un lokalizācijas algoritmu vāja vieta, kas prasa daudz aprēķinu pildot to parastajā procesorā, tāpēc tika veikta analīze par procesa paātrināšanu paralēlajās sistēmās.

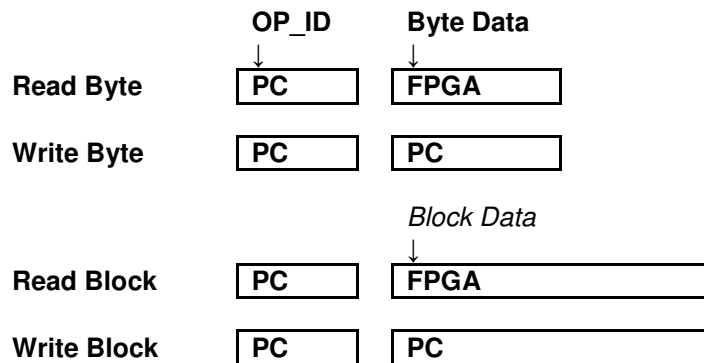
Turpmākie plāni:

- Eksperimenti ar esošo kameru bildes kvalitātes uzlabošanai;
- Bildes iegūšanas procesa paātrināšana, lielākas izšķirtspējas iegūšanai;

Biometrisku datu iegūšanas algoritmu paralelizācija un implementēšana programmējamos loģiskos masīvos

Datora saskarne ar prototipu

Iepriekš aprakstītā (personālā datora) PC saskarnes sistēma ar veidojamo prototipu (PROTO) tika pārveidota; iepriekš uz PROTO sūtāmās komandas principiāli dalījās 4 veidos: lasīt/rakstīt un katrā gadījumā — baitu/bloku.



Kur OP_ID saturēja ne tikai norādi ko darīt (lasīt/rakstīt) un cik daudz datu ir jāšūta/jāsaņem, bet arī izpildāmās operācijas būtību, piemēram (vairs netiek izmantots):

OP_ID:

7.	6.	5.	4.	3.	2.	1.	0.
B/BL	R/W	OPCODE					

B/BL: 1 — Byte Operation 0 — Block Operation

R/W: 1 — Write Operation 0 — Read Operation

OPCODE: uzdod veicamo operāciju (ko/kur tieši lasīt/rakstīt).

Šoreiz pieeja tika mainīta, pārveidojot PROTO saskarnes ar PC daļu par rakstāmo/lasāmo reģistru failu (baita operācijām) un daudzkanālu tiešās atmiņas bloka (Block Memory, jeb BM) piekļuves ierīci. Šim nolūkam "OP_ID.OPCODE" lauks tika aizvietots ar lasāmā/rakstāma baita/BM kanāla numuru, jeb adresi.

OP_ID:

7.	6.	5.	4.	3.	2.	1.	0.
B/BL	R/W	ADDR					

B/BL: 1 — Byte Operation 0 — Block Operation

R/W: 1 — Write Operation 0 — Read Operation

ADDR: defines either register address where to read/write (for byte operations), either block memory (BM) channel.

Veicot šīs izmaiņas, Manager_v3 bloka, kas veic saskarni ar datoru, uzdevums ir vienkāršojies, tāpēc, ka tagad tam ir iespējami tikai 4 scenāriji, kuri ir jāspēj apstrādāt:

1. **Baita lasīšana** (**B/BL = 1, R/W = 0; OP_ID = 0x80 + ADDR**)
2. **Baita rakstīšana** (**B/BL = 1, R/W = 1; OP_ID = 0xC0 + ADDR**)
3. **Bloka lasīšana** (**B/BL = 0, R/W = 0; OP_ID = 0x00 + Channel**)
4. **Bloka rakstīšana** (**B/BL = 0, R/W = 1; OP_ID = 0x40 + Channel**)

Tālākā BiTe PROTO dokumentācijā tiek minētas reģistru adreses un BM kanāli ar to nozīmi. Mēģinājums rakstīt/lasīt no neaizņemtā kanāla pagaidām rezultējas ar datu pazaudēšanu, vai sistēmas uzkāŗšanos.

Manager_v3 nenodarbojās ar ierakstīto datu glabāšanu, izņemot speciālus gadījumus: kā redzams, bloku operācijām šajā PROTO modifikācijā nekur netiek minēti rakstīto/lasīto bloku garumi un starta adreses – tāpēc tie tiek programmēti atsevišķi, izmantojot baitu operācijas, pirms izsaukt bloka operācijas. Šiem lielumiem tiek speciāli rezervēti sekojoši reģistri:

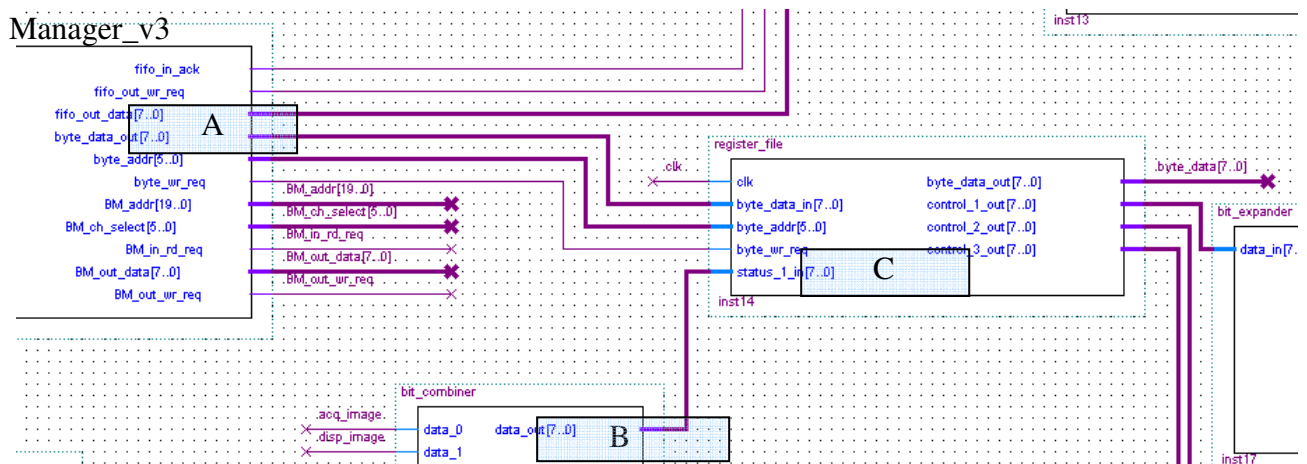
ADDR = 0x04

BM Operations Length* [7..0] (R/W)

ADDR = 0x05	BM Operations Length* [15..8] (R/W)
ADDR = 0x06	BM Operations Length* [“0000” & 19..16] (R/W)
ADDR = 0x07	Reserved, R/W Ignored, = 0x00
ADDR = 0x08	BM Operations Starting Address* [7..0] (R/W)
ADDR = 0x09	BM Operations Starting Address* [15..8] (R/W)
ADDR = 0x0A	BM Operations Starting Address* [“0000” & 19..16] (R/W)
ADDR = 0x0B	Reserved, R/W Ignored, = 0x00

**Piezīme: šajos reģistros pierakstāmas sākotnējās vērtības un bloku operāciju rezultātā šīs vērtības netiek izmainītas.*

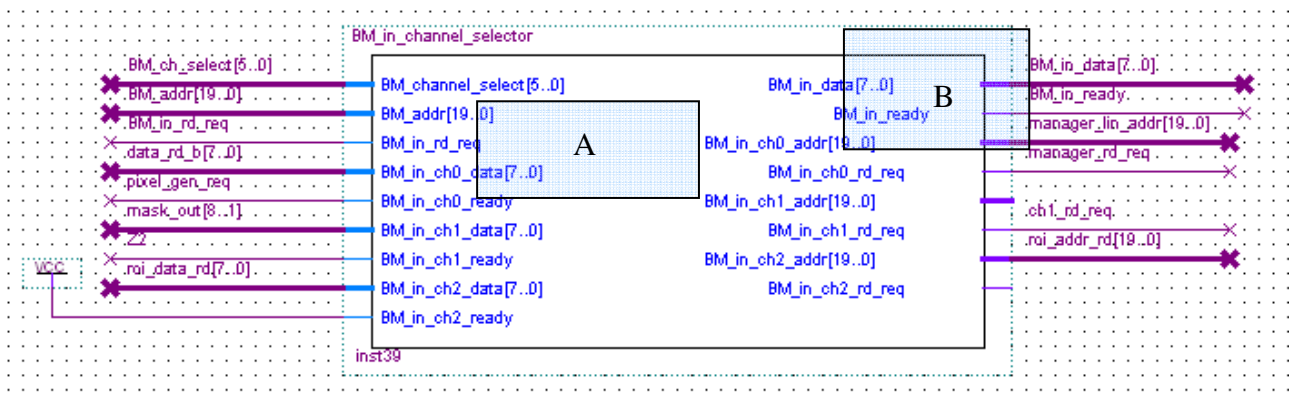
Darbs ar reģistru failiem norisinās caur adreses-datu-rakstīšanas pieprasījumu kopni (atzīmētas ar A), sekojoši:



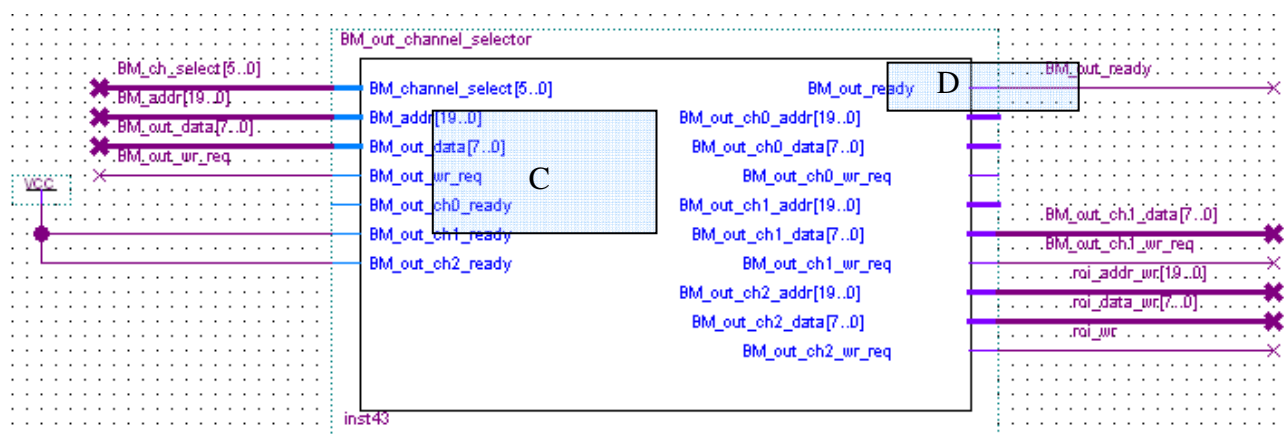
Par reģistru failu kalpo bloks `register_file`, kas ir pieslēgts pie `Manager_v3`. Reģistru failam realizēti 3 kontroles reģistri, kur caur “`bit_expander`” blokiem ir tālāk pieslēgtie pie attiecīgiem funkcionāliem mezgliem, kurus tie konfigurē. Bloks `bit_combiner` tiek izmantots ierīces stāvokļa nolasišanai, konkrēti šeit: vai tiek attēlots attēls, vai tiek iegūts attēls. Tikmēr uz datu kopnes `byte_addr[5..0]` mainās adrese, izvadā `byte_data[7..0]` parādās šajā reģistru šūnā ierakstītie dati, kuru var nolasīt ar Read Byte komandu.

Piezīme: pie datu līnijas signāliem A ir iespējams pieslēgt arī citus funkcionālus blokus, kuriem būs realizēti citi reģistri, galvenais, lai nesakristu reģistru adreses.

Lai viens `manager_v3` bloks spētu strādāt ar vairākiem atmiņas kanāliem, tiek izveidots BM kanālu sadalītais (demultipleksors):



un apvienotājs (multiplexors):



Ar A, B, C un D ir atzīmēti izvadi, kas ir paredzēti saskarnei ar Manager_v3 bloku:

- BM_channel_select uzdod to, ar kuru atmiņas kanālu, un līdz ar to arī ierīci, tiks strādāts,
- BM_addr uzdod adresi lasāmā/rakstāmā blokā,
- BM_**_data kalpo par lasāmo/rakstāmo datu kanāliem,
- BM_out_ready paziņo, ka ierīce, kas ir pieslēgta pie BM sadalītāja ir gatava datu saņemšanai no Manager_v3 bloka,
- BM_out_wr_req pieprasa BM ierīcei saņemt nākamo baitu no Manager_v3 bloka,
- BM_in_rd_req pieprasa ierīcei sagatavot nākamo baitu sūtīšanai uz Manager_v3 bloku,
- BM_in_ready paziņo, ka ierīce, kas ir pieslēgta pie BM apvienotāja ir gatava nodot datus Manager_v3 blokam.

Turpmākie plāni:

1. pilnveidot PC un PROTO saskarnes sistēmu;
2. realizēt normalizēto vektoru (ar automātisko jūtības pieskaņošanu) izdalīšanu no interesējošā reģiona (ROI), ar ROI parametriem uzdotiem no PC, izmantojot realizēto PC un PROTO saskarni.

Biometrisku datu kriptēšanas, glabāšanas un lietojama apakšsistēmas izveide

Biohash algoritmam tika apskatīta papildus informācija, ko pievienot datu vektoram, lai varētu uzlabot atpazīšanu. Pirms pievienošanas datu vektora izmērs bija aptuveni 512 parametri. Kā galvenās lietas, ko varētu pievienot, tika izvēlēti rezultāti no LBP (Local Binary Patter) un LDP (Local Derivative Patter). LBP pievieno 4196 parametrus, bet LDP pievieno 16784 parametrus. Veicot empīrisku testus, tika noskaidrots, ka, ja parametru daudzums pārsniedz 1000, tad dēļ ilgajiem izpildes laikiem, algoritms vairs nav praktiski pielietojami. Galvenā problēma ir ortogonālu matricu ģenerēšana. Līdz šim izmantotais algoritms bija Grama Šmita metode, kas ļauj no jebkuras matricas izveidot ortonormālu matricu. Lai to izdarītu, tika uzģenerēta gadījumskaitļu matrica ar nepieciešamo izmēru un pēc tam pielietota attiecīgā metode, lai to pārtaisītu ortonormālu. Algoritma izpildes sarežģītība bija aptuveni $O(n^3)$, kas

nozīmē, ka izpildes laiks pieaug kubiski atkarībā no ieejas datu daudzuma. Zemāk var redzēt nepieciešamos soļus, lai aprēķinātu rezultātu, kur „ v_k ” ir ieejas matricas k -tais vektors un „ e_k ” ir rezultējošās matricas k -tais vektors.

$$\begin{aligned}
 \mathbf{u}_1 &= \mathbf{v}_1, & \mathbf{e}_1 &= \frac{\mathbf{u}_1}{\|\mathbf{u}_1\|} \\
 \mathbf{u}_2 &= \mathbf{v}_2 - \text{proj}_{\mathbf{u}_1}(\mathbf{v}_2), & \mathbf{e}_2 &= \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} \\
 \mathbf{u}_3 &= \mathbf{v}_3 - \text{proj}_{\mathbf{u}_1}(\mathbf{v}_3) - \text{proj}_{\mathbf{u}_2}(\mathbf{v}_3), & \mathbf{e}_3 &= \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|} \\
 \mathbf{u}_4 &= \mathbf{v}_4 - \text{proj}_{\mathbf{u}_1}(\mathbf{v}_4) - \text{proj}_{\mathbf{u}_2}(\mathbf{v}_4) - \text{proj}_{\mathbf{u}_3}(\mathbf{v}_4), & \mathbf{e}_4 &= \frac{\mathbf{u}_4}{\|\mathbf{u}_4\|} \\
 & \vdots & & \vdots \\
 \mathbf{u}_k &= \mathbf{v}_k - \sum_{j=1}^{k-1} \text{proj}_{\mathbf{u}_j}(\mathbf{v}_k), & \mathbf{e}_k &= \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|}.
 \end{aligned}$$

Tika meklēts algoritms, kas varētu uzģenerēt matricu ātrāk nekā pašreizējais algoritms un vēlams arī ar mazāku sarežģītību. Lai to atrastu, tika meklēti algoritmi, kas dara ko līdzīgu.

Šī algoritma pamatideja ir QR dekompozīcija, kur no dotās matricas A ir iespējams izteikt divas matricas „ $A = QR$ ”, kur Q ir ortogonāla matrica, bet R ir augšējā trīsstūra matrica (visi elementi zem diagonāles ir vienādi ar 0). Izpētītajā literatūrā tika pieminētas trīs metodes:

- Gramma Šmita metode
- Householdera transformācija
- Izmantojot Givensa rotācijas

No abām jaunajām metodēm tika izvēlēta Givensa rotācijas metode. Oriģinālais algoritms noskaidro nepieciešamās transformāciju matricas, lai sākotnējo pārveidotu par augšējā trīsstūra matricu. Pēc tam transponētās transformāciju matricas savā starpā sareizina un kā rezultātu iegūst ortonormālu matricu. Katra transformāciju matrica ir Givensa matrica, kas spēj nonullēt izvēlētu elementu zem matricas diagonāles. Givena matrica izskatās sekojošā formā:

$$G(i, j, \theta) = \begin{bmatrix} 1 & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \dots & c & \dots & -s & \dots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \dots & s & \dots & c & \dots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 0 & \dots & 1 \end{bmatrix}$$

Kur atsevišķie elementi tiek aprakstīti sekojoši:

$$\begin{aligned}
 g_{kk} &= 1 & \text{for } k \neq i, j \\
 g_{ii} &= c \\
 g_{jj} &= c \\
 g_{ji} &= -s \\
 g_{ij} &= s & \text{for } i > j
 \end{aligned}$$

$$r \leftarrow \sqrt{a^2 + b^2}$$

$$c \leftarrow a/r$$

$$s \leftarrow -b/r.$$

Dotajās formulās vērtības a un b tiek izteiktas no apstrādājamās matricas.

No aprēķināšanas procesa var redzēt, ka svarīgākā sastāvdaļa ortonormālas matricas aprēķināšanai ir Givensa matrica. Nav nepieciešams gadījumskaitļu matricu pārveidot derīgā formā. Ir nepieciešams tikai uzģenerēt noteiktu daudzumu derīgu Givensa matricu un tās sareizināt ar identitātes matricu.

Kā rezultāts ir izveidots algoritms, kas lai iegūtu n*n izmēra ortonormālu matricu, sareizina (n*n / 2 – n) no gadījumskaitļiem ģenerētas Givensa matricas. Izpildes laiks ir samazināts salīdzinot ar Gramma Šmita metode. Cik var spriest no testiem tad nav samazināta algoritma izpildes laika sarežģītība, bet gan ir samazināta izpildes laika konstante.

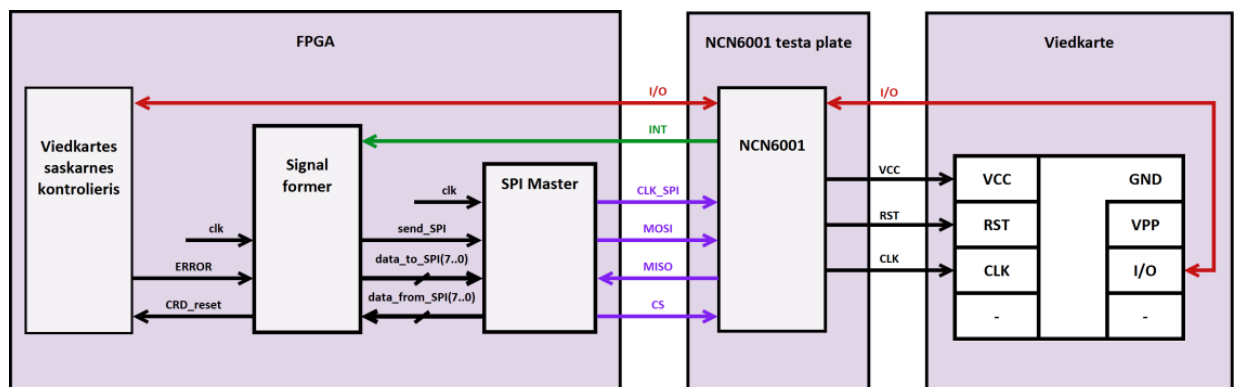
Viedkartes saskarnes izveide

Lai būtu iespējams nosūtīt datus uz viedkarti un saņemt atpakaļ atbildi no tās, bija nepieciešams izstrādāt FPGA programmējamajā loģikā implementētu moduli, kurš pilda viedkartes saskarnes kontroliera funkciju. Šī mērķa sasniegšanai tikai izmantots jau iepriekš izstrādāts shēmatēhniskais risinājums –NCN6001 integrālās mikroshēmas (IM) testa plate.

Pēc komandas, kas nosūtīta NCN6001 IM, izmantojot SPI (*Serial Peripheral Interface*) saskarni iespējams:

- Pieslēgt un atslēgt viedkartei barošanas spriegumu - V_{CC} ;
- Pieslēgt un atslēgt viedkartei atiestatīšanas – RST signālu;
- Nodrošināt viedkarti ar takts signālu - CLK;

NCN6001 IM arī dublē divvirzienu datu signālu I/O gan viedkartei, gan vadības iekārtai – šajā gadījumā FPGA programmējamajai loģikai. NCN6001IM arī ģenerē signālu \overline{INT} , ar kura palīdzību ir iespējams uzzināt to, vai viedkarte atrodas karšu lasītājā, vai nē (3.attēls).



3.attēls. NCN6001 testa plates un tās vadības signālu shēma

NCN6001 IM vadība FPGA realizēta sekojoši:

- 1)SPI saskarnes signāli tiek formēti un saņemti, izmantojot bloku „SPI Master”. Bloka uzdevums, saņemot *send_SPI* signālu, ir saformēt ieejas paralēlo signālu

data_to_SPI(7..0) izejas virknes signālā *MOSI (Master Output, Slave Input)*, bet SPI saskarnes ieejas signālu *MISO (Slave Output, Master Input)* pārveidot paralēlā izejas signālā *data_from_SPI(7..0)*.

2) Bloks „*Signal Former*” ne tikai ģenerē nosūtāmos SPI saskarnes signālus, bet arī pilda visa viedkartes saskarnes kontroliera vadības funkciju. Bloks strādā pēc šāda algoritma:

1. Kamēr viedkarte nav ievietota karšu lasītājā, visi pārējie saskarnes bloki atrodas noklusējuma stāvoklī. Datu nosūtīšana nav iespējama;
2. Ievietojot viedkarti karšu lasītājā tā tiek ieslēgta, veicot kartes atiestatīšanu – viedkarte nosūta saskarnes kontrolierim informāciju par tās specifiskajiem lietotajiem parametriem;
3. Notiek datu pārraide. Ja karte tiek izņemta no karšu lasītājā, ar signāla *CRD_reset* palīdzību tiek atiestatīti pārējie saskarnes kontroliera bloki noklusējuma stāvoklī. Ja viedkartes saskarnes kontrolieris ar signāla *ERROR* palīdzību paziņo, ka ir saņemta kāda kļūda – piem., viedkarte nereaģē, tad viedkarte tiek atiestatīta. Ja kļūdas ziņojums tiek saņemts 3 reizes pēc kārtas, viedkarte tiek izslēgta.

Viedkartes saskarnes kontrolieris implementēts nesaistīti ar NCN6001 vadības blokiem – neatkarīgi no tā, vai viedkarte ir vai nav ievietota karšu lasītājā, saskarnes kontroliera bloki jau gaida viedkartes atiestatīšanas atbildi – ja viedkarte ir pareizi ieslēgta, šo 2 līdz 33 baitus garo informāciju viedkarte nosūta saskarnes ierīcei (FPGA programmējamajai loģikai). Atiestatīšanas atbilde satur informāciju par viedkartes izmantotajiem specifiskajiem parametriem, to, kādu pārraides protokolu viedkarte izmanto u.c.

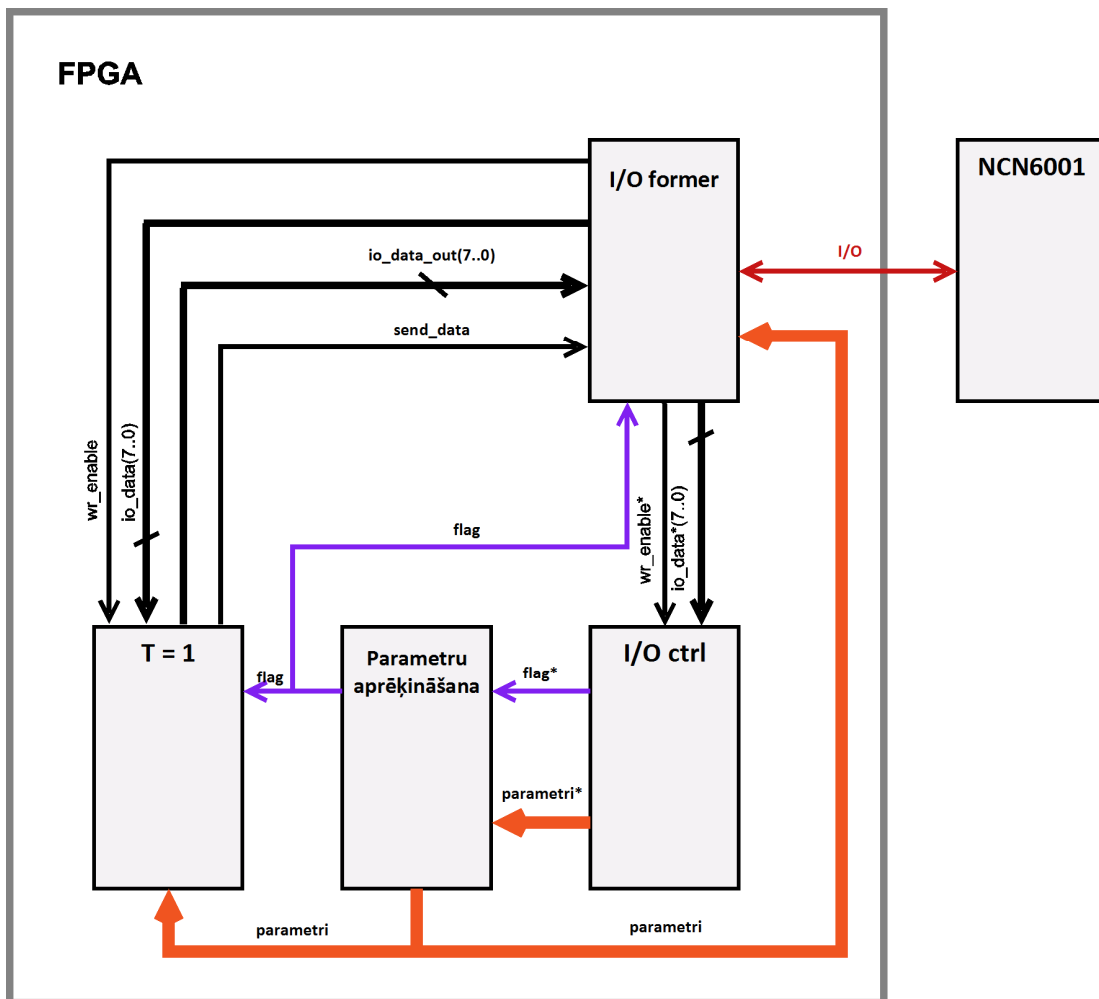
Atiestatīšanas atbildes saņemšanu raksturo 4.attēlā parādītā shēma. Tās veicamie uzdevumi:

1) Izmantojot bloku „*I/O former*” tiek saņemts un formēts divvirzienu datu signāls I/O. Pirms tiek uzsākta datu pārraide, viedkartes saskarnes kontrolierim ir jāsaņem atiestatīšanas atbilde no viedkartes, tādēļ bloks „*I/O former*” tiek ieslēgts datu saņemšanas režīmā. Saņemot atiestatīšanas atbildes baitus, bloks „*I/O former*” šo informāciju nosūta blokam „*I/O ctrl*”.

2) Bloks „*I/O former*” spēj nolasīt un „saprast” viedkartes atsūtīto atiestatīšanas atbildi. Bloks no tās izdala parametrus, kuri nepieciešami, lai realizētu tālāko datu nosūtīšanu un saņemšanu. Tiek izdalīti parametri, kuri raksturo to, cik takts signāla CLK ciklu laikā tiek nosūtīts viens bits pēc atbildes saņemšanas un cik ilgs ir gaidīšanas starplaiks starp 2 nosūtītiem baitiem vienā virzienā. Atiestatīšanas atbilde arī satur informāciju par viedkartes lietoto pārraides protokolu un tā specifiskajiem parametriem. Projektā tiek lietots T=1 protokols (šī protokola īpatnība ir tāda, ka saziņa ar viedkarti notiek, izmantojot informācijas blokus, t.i. mazākais nosūtāmais informācijas daudzums ir viens bloks, kurš sastāv no vairākiem bitiem), tādēļ tiek iegūti šī protokola parametri – maksimālais nosūtāmais informācijas baitu skaits viena bloka ietvaros, minimālais un maksimālais laika sprādis, kurā var tikt nosūtīti dati pretējā virzienā, kā arī maksimālais

laiks starp baitiem vienā virzienā. Kad visa atiestatīšanas atbilde ir saņemta, izdalītie parametri tiek nodoti blokam „Parametru aprēķināšana”.

3) Bloks „Parametru aprēķināšana”, atkarībā no tā, vai konkrētā viedkarte izmanto noklusējuma parametru vērtības, vai speciālās vērtības, kuru lielumi vai dati to aprēķināšanai saņemti ar atiestatīšanas atbildi, aprēķina konkrētās viedkartes izmantoto parametru vērtības. Kad parametru vērtības ir aprēķinātas, tās tiek nosūtītas blokam „I/O former” un blokam „T=1”, kurā implementēta daļa no LVS ISO/IEC 7816-3:2007 standartā noteiktās protokola T=1 funkcionalitātes. Ar signāla *flag* tiek pārslēgta bloka „I/O former” datu plūsma – tas vairs datus nesūta blokam „I/O ctrl”, bet gan datu apmaiņa notiek ar bloku „T=1”.



4.attēls. Atiestatīšanas atbildes saņemšanas realizācija FPGA

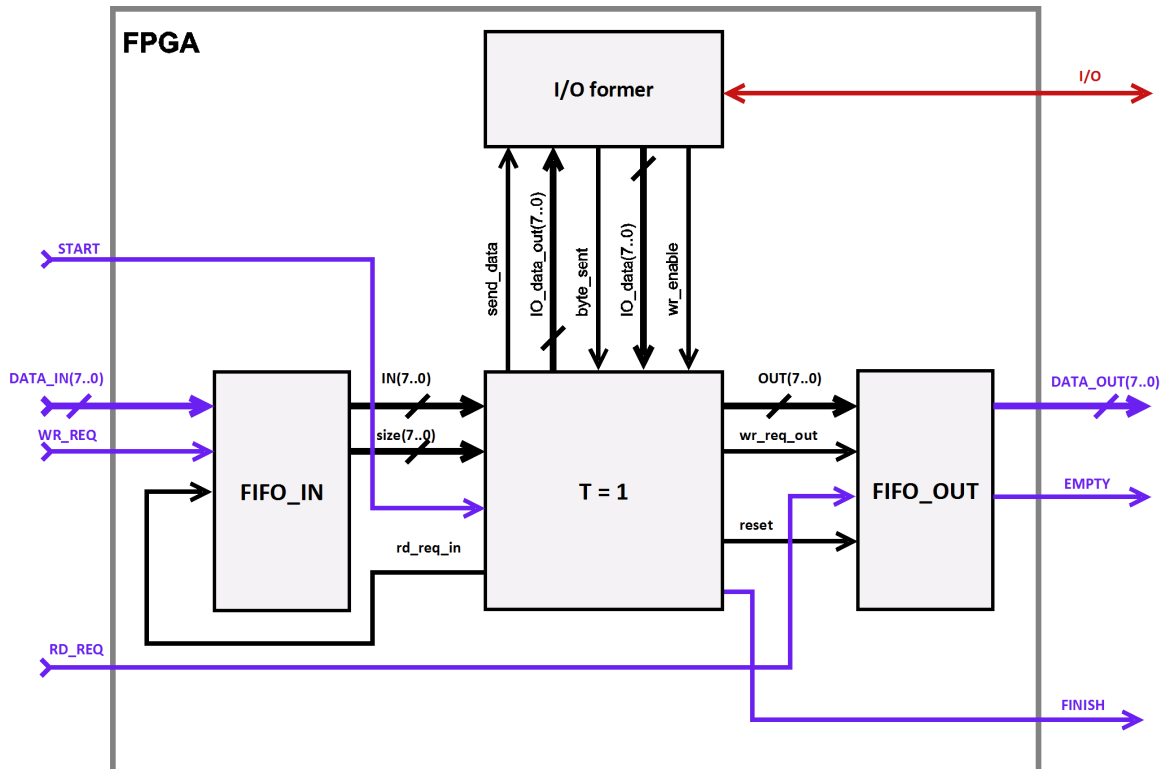
Informācijas pārraidi pēc atiestatīšanas atbildes saņemšanas raksturo 5.attēlā parādītā shēma. Datu nosūtīšana notiek šādi:

1) Saņemot augstu *flag* signāla līmeni (4.attēls) saskarni lietojošais bloks tiek informēts par to, ka saskarnes kontrolieris ir sagatavots darbam. Var sākties datu nosūtīšana un atbildes saņemšana.

2)Saskarni lietojošais bloks var iesūtīt nosūtāmo informāciju ieejas FIFO (*First In, First Out*) reģistrā „*FIFO_IN*” sev nepieciešamā ātrumā.

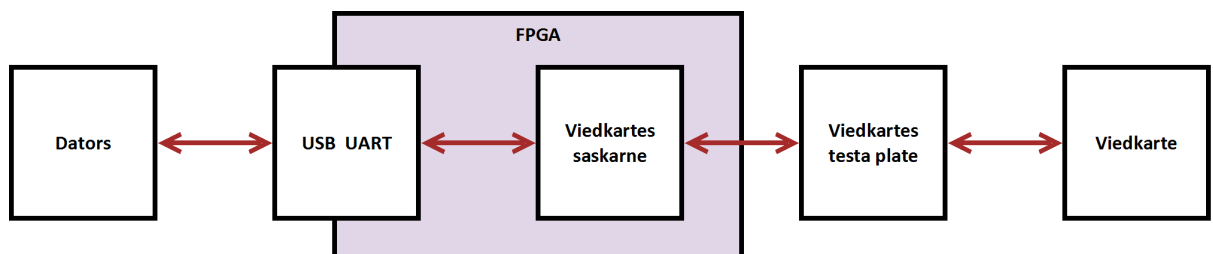
3)Kad informācija nosūtīta, var tikt padots *START* signāls. Saņemot šo signālu bloks „*T=1*” atbilstoši LVS ISO/IEC 7816-3:2007 standartā noteiktajā veidā saformē nosūtāmos datu blokus, un caur bloku „*I/O former*” tos nosūta viedkartei.

4)Pēc informācijas (komandas) nosūtīšanas viedkartei obligāti jāsniedz atbilde. Tā tiek izdalīta no viedkartes atsūtītā informācijas bloka, un ierakstīta „*FIFO_OUT*” izejas reģistrā. Kad visa informācija veiksmīgi saņemta, tiek ģenerēts signāls *FINISH*, ar kura palīdzību saskarni lietojošais bloks tiek informēts par to, ka dati nosūtīti.



5.attēls. Datu pārraides realizācija FPGA

Pašlaik tapšanas stadijā esošā viedkartes saskarne tiek testēta, izmantojot USB/UART (*Universal asynchronous receiver/transmitter*) moduli – ar šī moduļa palīdzību ir iespējama datu apmaiņa ar datoru – shematiski datu nosūtīšanas shēma parādīta 6.attēlā.



6.attēls. Datu nosūtīšanas shēma dators - viedkarte

Ir izstrādāta viedkartes aplikācija, kura dublē saņemtos datus – šādi ir iespējams notestēt vai korekti tiek saņemti un nosūtīti dati.

Turpmākie plāni:

1. izveidot viedkartes programmu, kas spētu komunicēt ar izstrādāto prototipu. Tiktu saņemti nešifrēti biometrijas dati vektoru formā, kas tiktu salīdzināti saglabātajiem datiem viedkartē. Atkarībā no tā vai dati ir pietiekami līdzīgi, tiek atgriezts pozitīvs vai arī negatīvs rezultāts;
2. uzsākt darbu pie biometrisko datu šifrēšanas moduļa FPGA, lai nebūtu jāsūta biimetriskie dati atklātā veidā;
3. turpināt testēt izveidoto saskarni, veikt koda kļūdu labošanu;
4. papildināt bloku „T=1” ar LVS ISO/IEC standartā noteiktajām protokola T=1 funkcijām, piem., dažādiem kļūdu labošanas mehānismiem.
5. risināt problēmu, kas saistīta ar NCN6001IM – pašlaik aptuveni 1-3% gadījumos, kad karte tiek ievietota karšu lasītājā un ieslēgta, NCN6001IM ģenerē nekorektu \overline{INT} signālu, kā rezultātā nav iespējama turpmāka saskarnes lietošana, kamēr NCN6001IM testa platei netiek atslēgts un no jauna pieslēgts spriegums.

Atsevišķu sistēmas komponentu un bloku izstrāde un izpēte

Šajā laika periodā svarīgākie veiktie darbi:

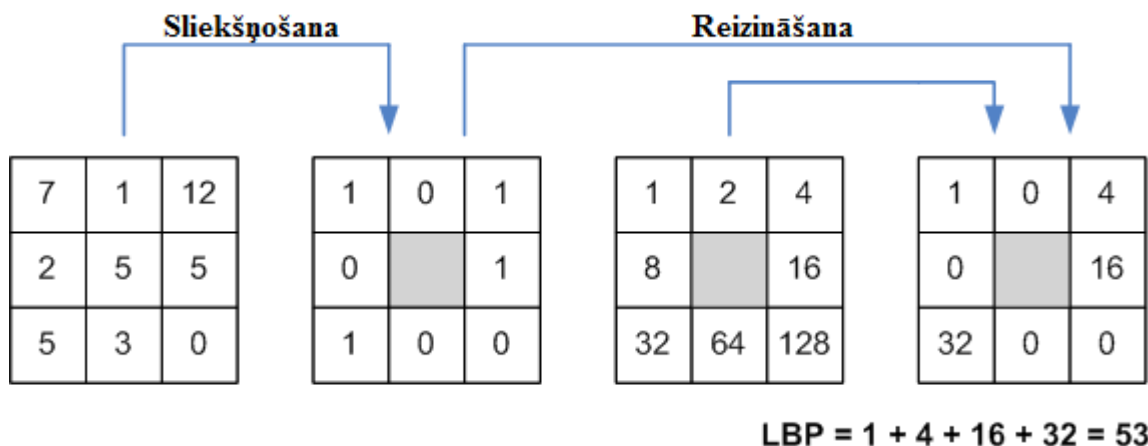
- MSP430 mikrokontroliera programmēšana;
- LBP, LDP algoritmu izpēte un realizācija MATLAB vidē;

Lai būtu iespējams uzņemt optimāli izgaismotus attēlus ir nepieciešams veikt attēlu sensora konfigurēšanu. Konfigurācijas parametri tiek iestatīti izmantojot I2C datu kopni, kura tika realizēta izmantojot MSP430F2003 mikrokontrolieri. Speciāli šim uzdevumam tika izveidota programma, kas pie ierīces ieslēgšanas uzstāda attēlu sensora sākotnējos konfigurācijas parametrus, kā arī saņemot attiecīgos vadības signālus no galvenā moduļa (FPGA) pārkonfigurē attēlu sensoru darbībai infrasarkanās vai redzamās gaismas diapazonā.

Šajā laika periodā tika veikta algoritmu Local Binary Patterns (LBP) un Local Derivative Patterns (LDP) izpēte un realizācija MATLAB vidē. LBP ir tekstūru aprakstošs operators, kas paredzēts pelēktoņu attēlu kaimiņpikseļu intensitāšu salīdzināšanai. Oriģināli operatora reģions ir 3x3, kur ir viens centrālais pikselis un 8 kaimiņpikseļi. Kaimiņpikseļu vērtības tiek slietšņotas pret centrālā pikseļa vērtību, tādā veidā iegūstot 8bitu bināro skaitli, vai tā decimālskaitļa ekvivalentu:

$$LBP_{P,R}(U_c) = \sum_{p=1}^P [s(Z_p) - Z_c] 2^{p-1}$$

kur $s(u) = 1$ ja $u \geq 0$ un 0 ja $u < 0$. Katrs rezultējošais decimālskaitlis ir uzskatāms par mikrostruktūru. Šīs mikrostruktūras bieži apraksta ar histogrammām. P ir kaimiņpikseļu skaits, bet R ir rādiusa izmērs.



Savukārt LDP operators izdala informāciju par intensitāšu izmaiņām vairākos virzienos pielietojot atvasināšanu. Katrs kaimiņpixselis sastāda kopējo kodu izmantojot atvasinājumu starp centrālo un kaimiņpixseli. Atvasinājumi tiek rēķināti vairākos virzienos, piemēram, 0, 45, 90 un 135 grādos. Attēla atvasinājumi tiek iegūti veicot kaimiņpixseļu atņemšanu, ņemot vērā virzienu kādā atvasinājumi tiek rēķināti.

$$I'_{0^\circ}(Z_0) = I(Z_0) - I(Z_4)$$

$$I'_{45^\circ}(Z_0) = I(Z_0) - I(Z_2)$$

$$I'_{90^\circ}(Z_0) = I(Z_0) - I(Z_2)$$

$$I'_{135^\circ}(Z_0) = I(Z_0) - I(Z_1)$$

kur Z0 ir centrālais pixselis un Z1, ..., Z8 kaimiņpixseļi.

Z1	Z2	Z3
Z8	Z0	Z4
Z7	Z6	Z5

Atkarībā no virziena atvasinājumu vērtības pret centrālo pikseli tiek kodētas katram kaimiņpixselim. Operāciju apraksta šāda sakarība:

$$f(I'_i(Z_0), I'_i(Z_i)) = \begin{cases} 0 & \text{if } I'_i(Z_0) - I'_i(Z_i) > 0 \\ 1 & \text{if } I'_i(Z_0) - I'_i(Z_i) \leq 0 \end{cases}$$

kur $i = 1, 2, \dots, 8$ kaimiņpixseļu indeksi. LDP kods sastāv no bitu virknes, tāpat kā LBP gadījumā.

Realizējot abus algoritmus MATLAB vidē, šīs funkcijas tika testētas izmantojot publiski pieejamo datubāzi, kas apstrādāta ar salāgto filtru ar Gausa masku. Tika iegūti sekojoši rezultāti:

LBP (R = 4, B = 16) CMC = 78%

LDP (R = 4, B = 16) CMC = 84 %

kur B ir bloku skaits, kuros tiek rēķināta histogramma pēc LBP vai LDP operācijas.

Tika secināts, ka LDP algoritms sniedz augstāku precizitāti nekā LBP algoritms. Lai uzlabotu atpazīšanas precizitāti tika izlemts veidot algoritmu, kas balstīts uz LDP un LBP pamatidejas, par kaimiņpixseļu izmantošanu, taču šoreiz par ieejas datiem tika ņemti Kompleksās salāgotās filtrācijas izejas dati – vektori. Izmantojot vektoriālo informāciju, ir iespējams aprēķināt katra vektora leņķi kādā tas ir vērsts, kā arī vektora garumu. Tika nolemts apstrādāt

vektoru absolūto garumu vērtības izmantojot LDP algoritmu, taču leņķiskās vērtības apstrādāt citādāk. Visi aprēķinātie leņķi tika normēti robežās no 0 – 180 grādiem. Tiek aprēķināta starpība starp centrālo leņķa vērtību un kaimiņ vērtībām. Ja starpība ir robežās ± 5 grādi, vērtība ir 1, ja lielāka tad 0. Šādā veidā tiek veidots jauns kods, kas ir rotācijas invariants, bet ņem vērā leņķu savstarpējo pagriezienu.

φ_1	φ_2	φ_3
φ_8	φ_0	φ_4
φ_7	φ_6	φ_5

$$LAP_{P,R}(U_c) = \sum_{p=1}^P [s(\varphi_p) - \varphi_0] 2^{p-1}$$

kur $s(u) = 0$ ja $u \geq 5$ un 1 ja $u < 5$. Leņķiskās starpības konstanti iespējams mainīt. Iegūtie rezultāti:

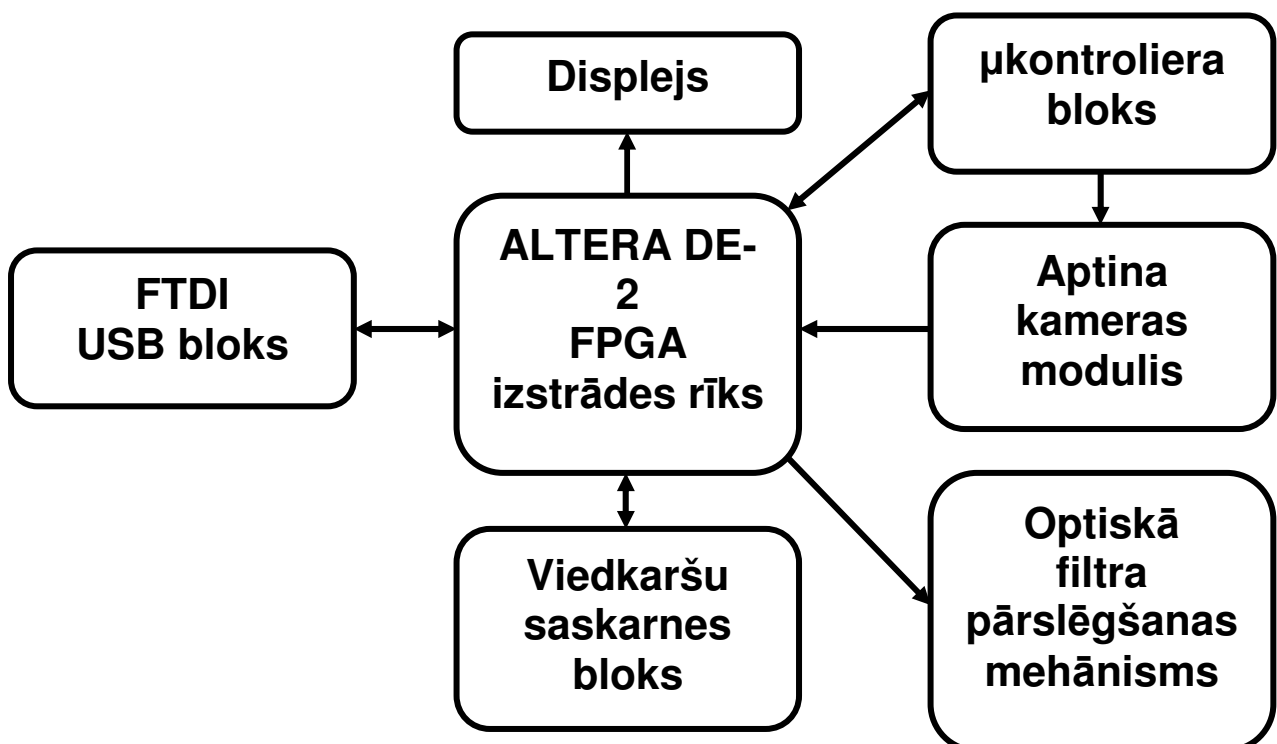
LAP (R = 4, B = 16) CMC = 89 %

Salīdzinot rezultātus redzams, ka apvienojot leņķisko informāciju ar vektoru intensitāti algoritma darbība uzlabojas par 5 %.

Turpmāk plānots veikt papildus eksperimentus ar jaunizveidoto datubāzi, kā arī papildināt algoritmu un novērtēt tā ātrdarbību.

Aktivitāte: Eksperimentālā izstrāde

Multimodālas biometrijas tehnoloģijas koncepta definēšana



7.attēls. Prototipa funkcionālo bloku shēma.

Prototips sastāv no vairākiem funkcionāliem blokiem, kas attēloti augstāk redzamajā attēlā. Katrs bloks pilda nozīmīgu funkciju kopējā prototipa darbībā. Prototipa galvenais funkcionālais bloks ir ALTERA DE-2 izstrādes rīks, kas nodrošina pārējo funkcionālo bloku sasaisti. Prototipa galvenais uzdevums ir spēt iegūt plaukstu attēlus divos dažādos gaismas spektra diapazonos – redzamajā gaismā un infrasarkanajā. Šī uzdevuma veikšanai ir izstrādāts Aptina kameras modulis, kas sastāv no Aptina MT9V024IA7XTM attēlu sensora, objektīvu turētāja ar filtru pārslēgšanās mehānismu, objektīva ar infrasarkanā spektra korekciju, baltās gaismas diodēm, infrasarkanās gaismas diodēm u.c. pasīvajiem komponentiem. Lai iegūtu optimāli eksponētu attēlu, attēlu sensoram nepieciešams iestatīt atbilstošus eksponēšanas parametrus. Šim nolūkam tika izveidots mikrokontroliera bloks, kas atkarībā no bildes, kāda tiks uzņemta, nodod eksponēšanas parametrus attēlu sensoram (pārprogrammē attēlu sensoru). Mikrokontroliera bloks sastāv no Texas Instruments MSP430F2013 mikrokontroliera, kas tiek programmēts izmantojot datoru. Saikne starp ALTERA DE-2 izstrādes rīku tiek nodrošināta lai mikrokontrolieris zinātu kādi konfigurācijas parametri ir jāieprogrammē attēlu sensora reģistros. Papildus eksponēšanas parametriem ir nepieciešams pareizos laika momentos pārslēgt optiskā filtra mehānismu no redzamās gaismas filtra uz infrasarkanās gaismas filtru. Šo darbību veic optiskā filtra pārslēgšanas bloks, kas ir realizēts ar H-tilta mikroshēmu un tiek vadīts no ALTERA DE-2 izstrādes rīka. Lai attēlus varētu iegūt datorā, prototipa izstrādes gaitā tika realizēts arī FTDI USB bloks, kas nodrošina prototipa un datora saskarni izmantojot USB datu kopni. USB datu kopne var tikt izmantota arī lai kontrolētu prototipa darbību, kā arī iegūtu attēlus. Papildus, prototipam tika pievienots viedkaršu saskarnes bloks, kas tiks izmantots lai nosūtītu biometrijas datus uz viedkarti un saņemtu rezultātu par datu līdzību. Viedkartē tiek veikta iepriekš saglabāto datu salīdzināšana. Papildus prototipam tika pievienots arī displejs, kas nodrošina iegūto attēlu attēlošanu, kā arī var tikt konfigurēts kā skārienjūtīga virsma lietotāju interfeisa izveidei.

Tehnoloģijas demonstratora eksperimentālā maketa montēšana

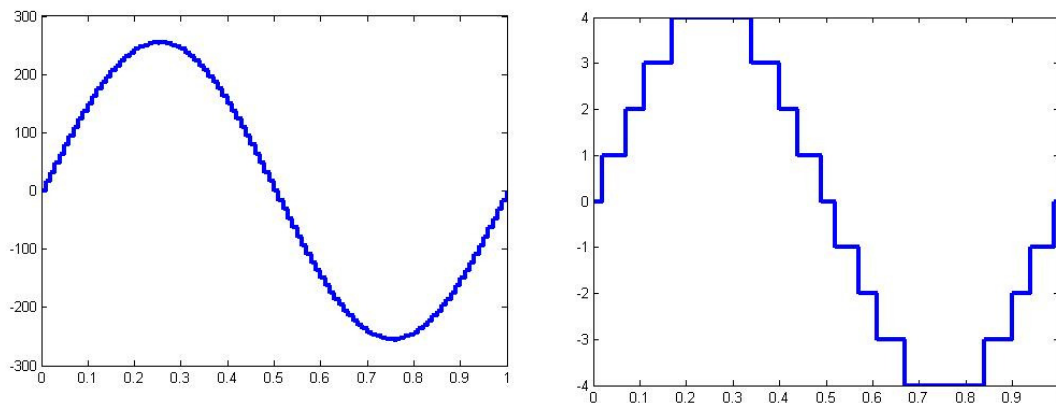
Šajā laika periodā ir veikta biometrijas prototipa 1.versijas montāža, kā rezultātā ir savienoti galvenie sistēmas bloki: FPGA izstrādes rīks, attēlu sensora modulis; optiskā filtra pārslēgšanās mehānisma shēma; viedkaršu modulis; šķidro kristālu displejs. Visi šie sistēmas bloki ir apvienoti un ievietoti organiskā stikla ietvarā, veidojot vienotu prototipēšanas sistēmu. Izmantojot šo sistēmu ir izveidota plaukstu nospiedumu un asinsvadu attēlu datubāze ar 100 cilvēku paraugiem. 8. attēlā ir redzama izveidotā biometrijas prototipa iekārta.



8.attēls. Biometrijas prototipa iekārta.

Algoritmu implementēšana eksperimentālajā maketā

Tika veiktas modifikācijas CMF, nomainīts maskas izmērs no 9x9 uz 15 x15, kā arī tika izmainīta maska. Līdz šim bija izmantota maska, kuras norma bija vienāda ar 1, lai pēc filtrācijas iegūtais rezultāts nebūtu jāreizina ar normalizējošu koeficientu. Taču tika atklāts, ka šādai pieejai, filtra maskas aprakstīšanai netika izmantoti visi iespējamie kvantēšanas līmeņi, līdz ar to radās kvantēšanas kļūdas, kas noveda pie tā, ka maska tika kropļota – vairs nav lēzena, bet kāpņveidīga.

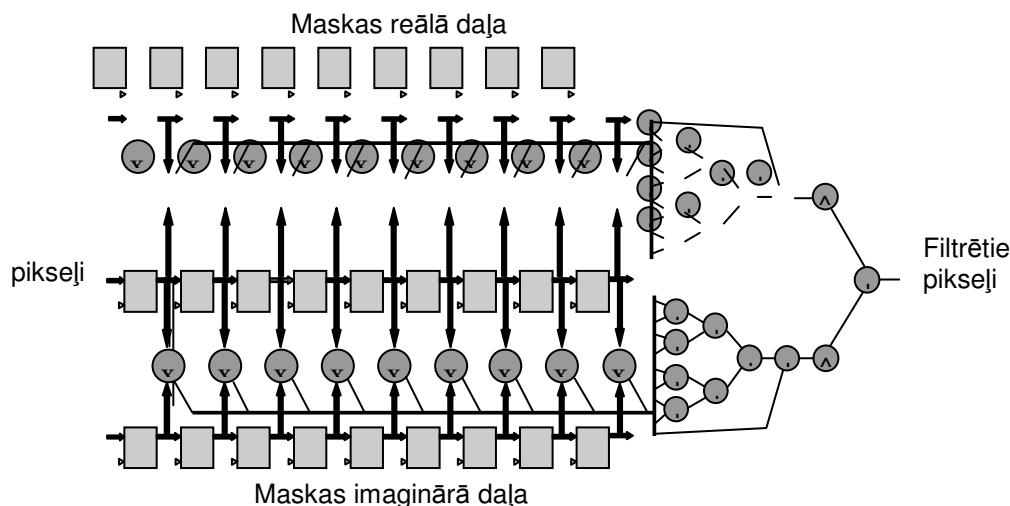


9.attēls. Maskas piemēri

Normalizēta maska tika aizvietota ar maksimālas izšķirtspējas masku, proti, filtrācijas rezultāts tagad ir jānormē filtra izejā. Izmantojot maksimālo bitu daudzumu maskas kropļojumi tiek samazināti.

Šajā laikā tika gūts priekšstats par parametrizējamām ķēdēm VHDL. Tas noderēja veidojot jauno CMF versiju.

Pagaidām tiek veidotas atsevišķas CMF (kas ir redzams 10. attēlā) parametrizējamās sastāvdaļas, tādas kā *Maskas reālā daļa*, *Maskas imaginārā daļa*, *reizinātāji* un *summatori*. Piemēram, *Maskas reālā daļa* sastāv no virknē slēgtiem DFF trigeriem, to skaits ir atkarīgs no maskas izmēra. Parametrizācijas priekšrocība ir tāda, ka ir iespējams mainīt sistēmas parametrus, nemodificējot izveidoto VHDL kodu, bet mainot tikai parametru vērtības.



10.attēls CMF blokshēma

Filtra stabilitātei ir nepieciešams izveidot konveijera arhitektūru, jeb pēc katras aritmētiskās operācijas, ko pilda katra filtra sastāvdaļa, rezultāts jā saglabā reģistros un tālāko apstrādi jāveic nākamajā sistēmas pulksteņa (clock signal) taktī. Šo pieeju sauc par "pipeline", pielietojot to, katrai filtra sastāvdaļai ir pietiekams laiks, lai veiktu nepieciešamus aprēķinus. Izmantojot šādu arhitektūru oriģinālie (nefiltrētie) un filtrētie dati tiks aizkavēti par tik takts signāla periodiem, cik ir izmantoti reģistri, konveijera arhitektūras nodrošināšanai, kas saglabā aprēķinu starpvērtības.

Turpmāk paredzēts uzsākt Ātrā Bez-Halo (NH-CMF2) filtra realizāciju.

Secinājumi

Projekta „BiTe” pētnieciskā darbība „Lietišķo pētījumu” aktivitātē sekmīgi turpinās arī piektajā periodā. Noris izstrādājamās tehnoloģijas atsevišķu moduļu, algoritmu un elektrisko principiālo shēmu izpēte un izveide. Šajā pārskata periodā sekmīgi uzsākta arī „Eksperimentālās izstrādes” aktivitāte.

Attēlu sensora iespiedplates projektēšanai un attēlu iegūšanas sistēmai izmantojot FPGA tiek plānots izstrādāt MATLAB kodu ģeometrisku raksturlielumu kopu identifikācijas algoritma blokshēmai, turpināt apskatīt ģeometrisku raksturlielumu algoritmus un attēlu iegūšanas metodes.

Sejas atpazīšanas sistēmai tiek plānoti eksperimenti ar esošo kameras moduli, bildes kvalitātes uzlabošanai un iegūšanas procesa paātrināšanai, kā arī attēla izšķirtspējas palielināšanai.

Biometrisko datu iegūšanas algoritmu paralelizācijas un implementēšanas programmējamos loģiskos masīvos aktivitātes ietvaros nākotnē paredzēts pilnveidot PC un PROTO saskarnes

sistēmu un realizēt normalizēto vektoru no ROI, ar ROI parametriem uzdotiem no PC, izmantojot realizēto PC un PROTO saskarni.

Biometrisko datu kriptēšanas, glabāšanas un lietojama apakšsistēmas izveides aktivitātē paredzēts izveidot viedkartes programmu, kas spētu komunicēt ar izstrādāto prototipu; uzsākt darbu pie biometrisko datu šifrēšanas moduļa FPGA; turpināt testēt izveidoto saskarni; papildināt bloku „T=1” ar LVS ISO/IEC un nekorekta **INT** signāla problēmas novēršanas risināšana.

Atsevišķu sistēmas komponentu un bloku izstrādei un izpētei nepieciešams veikt papildus eksperimentus ar jaunizveidoto datubāzi, kā arī papildināt algoritmu un novērtēt tā ātrdarbību. Multimodālas biometrijas tehnoloģijas koncepta definēšanas aktivitātē aprakstīto prototipa funkcionālo blokshēmu tuvākajā laikā nav plānots mainīt, bet par visās veiktajām izmaiņām tiks aprakstīts nākošajos progresa pārskatos.

Tehnoloģijas demonstratora eksperimentālā maketa montēšanas apakšaktivitātē maketa pirmās versijas detalizēts izveides apraksts ir sniegts šajā progresa pārskatā. Turpmāk veiktie iekārtas papildinājumi un uzlabojumi tiks aprakstīti nākošajos progresa pārskatos.

Algoritmu implementēšana eksperimentālajā maketā notiks arī turpmāk, nākotnē paredzēts uzsākt darbu pie Ātrā Bez-Halo (NH-CMF2) filtra realizācijas.

Projekta pētnieciskie rezultāti tiek apspriesti izpildītāju iknedēļas sanāksmēs. Šīs sanāksmes tiek veidotas, lai projekta izpildītāji kopīgi izdiskutētu par pētījumiem un problēmām, kas jārisina, lai aktivitātes veiksmīgi attīstītos. Piektajā progresa pārskata periodā notika sanāksmes, kurās tika pārrunāti dažādi jautājumi saistībā ar projektu, svarīgākie no tiem: CMF filtra parametru izmaiņas; Rīda-Solimona koda kļūdu labošana; LBP implementācija signālprocesorā; demonstratora izstrāde; datubāzes uzņemšana, ortonormālas matricas ģenerēšana, u.c.

Iepriekšējā laika periodā tika sagatavots zinātniskais raksts par projektā veiktajām aktivitātēm, kas saistītas ar biometrijas datu iegūšanu, apstrādi un šifrēšanu. Raksts „Biobatching and Fusion of Palmprint and Palm Blood Vein Biometric Data” prezentēts starptautiskā IEEE organizētā konferencē par plaukstas biometrisko parametru izmantošanu personu atpazīšanā (ICHB2011), Honkongā.

Turpinās darbs pie aktivitātes „Intelektuālā ģeometriskās formas aizsargāšana”, līguma ar patentpilnvaroto SIA „PĒTERSONA PATENTS” ietvaros ir sagatavots un iesniegts starptautiska patenta pieteikums.

Ir noteikti nākamā perioda pētniecisko darbu uzdevumi un sasniedzamie rezultāti. Par projekta pētnieciskiem sasniegumiem tiek plānots informēt arī piedaloties ar referātiem un publikācijās konferencēs un semināros.

Kopu identifikācijas algoritma blokskhēma:

