VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS KATEDRA

Vizualiosios kriptografijos algoritmai

Algorithms for Visual Cryptography

Magistro baigiamasis darbas

Atliko: **Povilas Rastenis**

Darbo vadovas: **Dr. Vilius Stakėnas**

Recenzentas: **Dr. Gintaras Skersys**

Vilnius 2016

Turinys

Anotacija

Darbe nagrinėjami vizualiosios kriptografijos algoritmai. Apžvelgiami klasikinis ir tikimybiniai algoritmai, vizualioji kriptografija ir atsitiktinės gardelės. Tiriami algoritmai dvispalvių vaizdų šifravimui, algoritmai šifruoti kelioms paslaptims. Pasiūlyti nauji spalvotų vaizdų kriptografijos algoritmai ir jų patobulinimai. Atliekamas saugumo tyrimas.

Raktiniai žodžiai: vizualioji kriptografija, algoritmai, atsitiktinės gardelės, šifravimas

Summary

This paper deals with visual cryptography algorithms. An overview of classical and probabilistic algorithms, visual cryptography and random grids is presented. Tested algorithms for encryption of binary images, algorithms for encryption of a few secrets. New algorithms for cryptography of color images was presented and improvements were offered. Developed and improved algorithms for color images cryptography. Safety study of algorithms is presented

Keywords: visual cryptography, algorithms, random grids, encryption

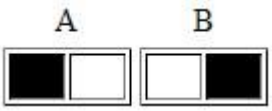
1. Įvadas

Kriptografija – tai mokslas apie informacijos apsaugą. Vienas iš jos uždavinių - informacijos prasmės ar struktūros slėpimas. Tai atliekama šifruojant informaciją. Kriptografija buvo aktuali ir anksčiau, tačiau ta labai aktualu ir šiandien. Viena iš kriptografijos mokslo šakų yra vizualioji kriptografija. Pats vizualiosios kriptografijos pavadinimas kilęs iš jos šifravimo principo – jei slapto vaizdo šifravimui reikalingi skaičiavimo algoritmai, tai dešifravimas atliekamas žmogaus regos sistemos pagalba. Ši kriptografijos šaka aktuali ir dėl to, kad dešifravimas gali būti atliekamas bet kokioje aplinkoje, netgi neturint kompiuterio. Ji išsiskiria saugumo ir dešifravimo paprastumo santykiu. T. y. dešifravimas lengvas, o šifras labai saugus. Dėl tokio paprasto dešifravimo susitaupo laiko, nereikia vykdyti jokių veiksmų, kurie atimtų daug laiko, tiesiog skirtingas dalis sudedame vieną ant kitos ir pamatome atsakymą. Vizualioji kriptografija yra naudojama priėjimo kontrolėje, autoriaus teisių apsaugoje, elektroniniuose parašuose, elektroninėje valiutoje, virtualiuose rinkimuose, vandens ženkluose, vaizdinėje autentifikacijoje ir identifikacijoje [CY12].

Svarbią kriptografijos sritį sudaro kriptoanalizės uždaviniai. Kriptoanalizė – tai kriptografinių apsaugos algoritmų patikimumo vertinimas. Tai gali pasirodyti kaip natūrali kriptografijos priešingybė, tačiau dažnai šios sritys yra viena kitą papildančios – geras kriptoanalizės supratimas leidžia kurti saugesnius kriptografijos metodus. Vizualiojoje kriptografijoje šifruoti tašką galima keliais būdais. Šifro saugumas priklauso nuo šių būdų pasirinkimo tikimybių. T. y. kuo didesnė tikimybė, kad taškai bus užšifruoti vienodai, tuo didesnė tikimybė juos atskleisti.

Mano darbo tikslas išnagrinėti vizualinės kriptografijos algoritmus ir pasiūlyti patobulinimų.

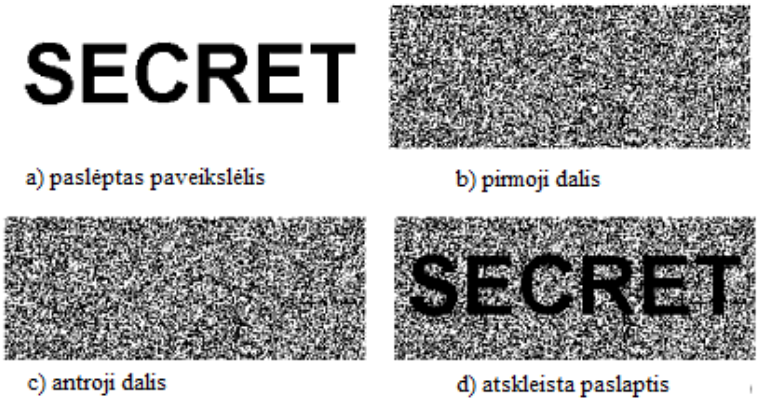
Klasikiniame vizualiosios kriptografijos modelyje naudojamas paslapties padalijimo metodas. Paprasčiausiu atveju šifruojamas vaizdas yra skaidomas į dvi atskiras sudedamąsias dalis. Kai abi dalys perdengiamos, išryškėja slaptas vaizdas. Pavyzdžiui, tegu informacija yra tiesiog „baltas" taškas. Kaip raktą parinkime figūrą A (arba B), šifras irgi bus A (arba B).



Pav. 1. Figūros A ir B

Jeigu šios figūrėlės atspausdintos ant skaidrios plėvelės, tai uždėję vieną ant kitos (figūrą A ant figūros A arba figūrą B ant figūros B) ir žiūrėdami iš tolo matysime pilką tašką. Tai reiškia, kad buvo užšifruotas baltas taškas. Jeigu reikia užšifruoti juodą tašką, tai kaip raktą parinkime A (arba B), o šifras bus B (arba A). Tada uždėję abi figūras vieną ant kitos matysime juodą tašką.

Kiekvienas paveikslėlis susideda iš daug taškų. Todėl norint užšifruoti bet kokį paveikslėlį, šią procedūrą reikia atlikti kiekvienam jo taškui. Pavyzdžiui (Pav. 2) [WY12]:



Pav. 2. 2 iš 2 schema

Šis modelis gali būti išplėstas iki paslapties dalijimo su slenksčiu k varianto, šiuo atveju šifruojamas vaizdas yra skaldomas į n dalių ir slaptas vaizdas matomas tik tada kai viena ant kitos yra sudedamos k (ar daugiau) dalių. Pavyzdžiui (Pav. 3) [WY12]:

Pav. 3. 2 iš 3 schema

Klasikiniai modeliai gali šifruoti tik dvispalvius paveikslėlius, todėl jei paslaptis turi daugiau spalvų (pvz.: pilką spalvą), reikia naudoti tonavimo algoritmus ir taip paversti paslaptis į dvispalvius paveikslėlius. Be šių klasikinių modelių yra ir kiti, todėl apžvelgus su pasirinkta magistrinio darbo tema susijusią literatūrą, priimtas sprendimas labiau įsigilinti į vizualiosios kriptografijos algoritmus su atsitiktinėmis gardelėmis (*angl.* random grids). Apžvelgta literatūra apėmė tris esmines sritis: klasikinį vizualiosios kriptografijos modelį, tikimybinius vizualiosios kriptografijos algoritmus ir algoritmus su atsitiktinėmis gardelėmis.

Nagrinėti algoritmai:

- Du algoritmai dvispalviams paveikslėliams;
- Tonavimo algoritmai;
- Du algoritmai trispalviams paveikslėliams;
- Dviejų paslapčių šifravimo algoritmas;
- Algoritmas šešių spalvų paveikslėliams.

Visiems algoritmams atlikta literatūros apžvalga, realizuotos programos. Sugalvoti algoritmai trispalviams paveikslėliams, jie patobulinti iki šešių spalvų paveikslėlių šifravimo. Keičiant taškų šifravimo būdų pasirinkimo dažnį tirtas saugumas ir paslapties atskleidimo galimybė neturint visos informacijos.

2. Vizualiosios kriptografijos metodų apžvalga

2.1. Klasikinis vizualiosios kriptografijos modelis

Optimalaus kontrastingumo problema yra tiriama nuo pat vizualiosios kriptografijos sukūrimo. Ieškoma optimalaus potaškių[1] skaičiaus su kuriuo atskleista paslaptis būtų lengvai atpažįstama. Kuo ryškiau matosi paslaptis tuo didesnis kontrastingumas.

Taškams yra įvedami kontrastingumo slenksčiai *l* ir *h* [CY12]. Atkuriamas taškas laikomas baltu, jei jis turi pakankamai baltų potaškių, t. y. juodų potaškių skaičius yra mažesnis arba lygus slenksčiui *l*. Ir taškas laikomas juodu, jei jis turi pakankamai juodų potaškių, t. y. daugiau arba lygiai slenksčiui *h*. Pateikiama kontrastingumo formulė $\gamma = (h - l)/m$, kur m yra potaškių skaičius.

Hofmeister [HKS00] optimalaus kontrastingumo radimo problemai pasiūlė tiesinį sprendimo būdą. Aptariamas kontrastingumo ir potaškių santykis. **Lentelė 1** [HKS00] parodo suskaičiuotą optimalų kontrastingumą keliems *k* iš *n* schemos variantams.

*k**n*	2	3	4	5	6	...	10	...	50	...	100
2	1/2	1/3	1/3	3/10	3/10		5/18		25/98		25/99
3		1/4	1/6	1/8	1/10		1/12		13/196		625/9702
4			1/8	1/15	1/18		1/35		1161/65800		425/25608

Lentelė 1

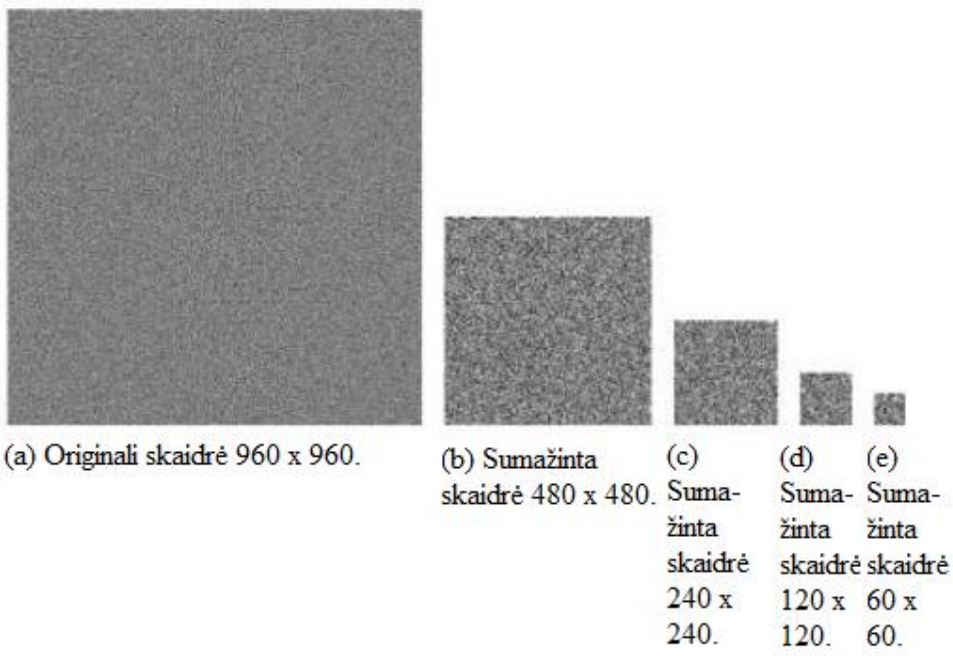
Efektyvesnei vizualiajai kriptografijai siūloma naudoti XOR operaciją. [THLT05] Šiuo atveju nebeliks tradicinio skaidrių sudėjimo į vieną, tai pakeistų XOR. Ši schema pasižymi gera rezoliucija ir aukštu kontrastingumu.

Klasikinėje vizualiojoje kriptografijoje naudojami balti ir juodi taškai, juos galime žymėti 0 ir 1. Šie balti ir juodi taškai nesikeičia. Balti taškai visada bus balti, o juodi taškai visada bus juodi. Taškų reikšmės lieka tos pačios ir po to kai paveikslėlis yra keičiamas.

Dvejetainiai paveikslėliai yra atsparūs dažnai naudojamoms atakoms. Tos atakos yra: paveikslėlio dydžio keitimas, karpymas, mastelio keitimas, iškraipymas ir suspaudimas. Po tokių atakų balti taškai išlieka balti, o juodi išlieka juodi. Kadangi šie taškai kitokių reikšmių įgyti negali, dvejetainiai paveikslėliai puikiai tinka saugoti tam tikrus duomenis.

Mastelio keitimas, karpymas ir suspaudimas dažnai naudojami bandant pamatyti paslaptį. Pav. 4 [WY12] parodo skaidrę, kuri buvo kelis kartus sumažinta. Akivaizdu, kad jokios informacijos susijusios su paslaptimi nematyti. Šie paveikslėliai taip pat buvo sumažinimo metu suspausti, tai patvirtina, kad suspaudimas skaidrių neįtakoja.

[1] Angl. subpixel – padalinto taško dalis. LCD technologijoje vienas taškas dažniausiai yra padalijamas horizontaliai į tris dalis.



Pav. 4 Skaidrės mažinimas

Vizualiosios kriptografijos saugumas, kaip ir daugelyje kriptografinių schemų, remiasi atsitiktinumu. Tiksliau, kuriant skaidrę, kiekvieno taškas turi vienodą tikimybę tapti tiek juodu, tiek baltu . Pavyzdžiui, pirmoji schema sukurta Naor ir Shamir [NS95] taško struktūrą rinkdavosi atsitiktinai. Toliau, reikėjo nuspręsti koks, juodas ar baltas, taškas iš paslapties bus atvaizduojamas šiomis struktūromis. Jei reikia atvaizduoti juodą tašką, atsitiktinai pasirinkamos atitinkamos struktūros. O jei reikia atvaizduoti baltą tašką, atsitiktinai pasirenkama viena struktūra ir patalpinama į abi skaidres.

Tai yra pagrindinis vizualiosios kriptografijos saugumo požymis. Tai reiškia, kad jokia kriptografinė analizė neatskleis paslapties, jei nagrinės tik vieną skaidrę.

Ši idėja gali būti patikrinta naudojant tobulą slaptumą (angl. *perfect secrecy*), kurį pirmą kartą pristatė Shannon [Sha49]. Pagal Shannon šifrų sistemos apibrėžimas yra:

1. Šifrų sistema yra *n*-vietis sąrašas *T*, kuriame yra transformacijos iš žinučių aibės M į šifrų aibę C.
2. Kiekvienam $t_i \in T$ yra priskirta p_i, kur p_i yra tikimybė, kad bus pasirinkta t_i.
3. Taip pat kiekviena žinutė turi savo tikimybę.

Taigi, pasirinkus klasikinį *k* iš *n* vizualiosios kriptografijos modelį, kur $k = n = 2$, galima lengvai pamatyti panašumą į one-timepad algoritmą[2]. Viena skaidrė yra šifras, o kita – raktas. Tai primena one-timepad algoritmą, nes kiekvienas šifro skaidrės taškas yra dekoduojamas jam

[2] Teksto šifravimo algoritmas, kuriame šifras gaunamas teksto simbolius sumaišius su rakto simboliais. Plačiau: http://en.wikipedia.org/wiki/One-time_pad

ekvivalenčiu tašku rakto skaidrėje. Tai pagrindžia vizualiosios kriptografijos saugumą, tačiau [WY12] šaltinio autoriai pateikia ir formalų įrodymą:

Teiginys. *2* iš *2* schemos dalijimasis paslaptimis yra saugus.

Įrodymas. Tarkime turime (2,2) schemą dvejetainiams paveikslėliams. Žinučių aibė susidarys iš dviejų elementų, taško spalvos: baltos (0) ir juodos (1),

$$M = \{0,1\}.$$

Paprastumo dėlei, m_0 ir m_1 žymėsime taško reikšmę, 0 ir 1 atitinkamai.

Yra šeši būdai, kaip kiekvienas taškas bus užšifruotas, todėl turime:

$$C = \{[1,1,0,0], [0,0,1,1], [1,0,0,1], [0,1,1,0], [1,0,1,0], [0,1,0,1]\}$$

Įvykį, kad skaidrė iš keturių potaškių yra j – osios struktūros iš aibės C žymėsime c_j. Akivaizdu, kad $0 \leq j \leq 5$ ir c_j yra vienodai tikėtini. Galime manyti, kad atsitiktinai pasirinkto slapto paveikslėlio taškai yra tolygiai pasiskirstę, todėl $p(m_0) = p(m_1) = 0,5$. Paimkime vieną skaidrę. Visiems j struktūra c_j gali būti sudedama su tokia pat antros skaidrės struktūra c_j, kad gauti baltą (pilką) tašką, arba ji gali būti sudedama su priešinga struktūra $c_j + 1$ (jei j lyginis) ir $c_j - 1$ (jei j nelyginis), kad gauti juodą tašką. Kitaip tariant, yra vienoda tikimybė, kad taškas bus juodas arba baltas, todėl kiekvienam j, $p(m_0|c_j) = p(m_1|c_j) = 0,5$.

Taigi, kiekvienam i ir j, gausime $p(m_i|c_j) = p(m_i) = 0,5$, tai užbaigia tobulo šifro įrodymą. Šis įrodymas parodo, kad ši vizualiosios kriptografijos schema yra pakankamai saugi, kad būtų galima naudoti praktikoje.

Daugumos vizualiosios kriptografijos schemų skaidrių dydžiai gali labai išaugti, tai priklauso nuo paveikslėlio tipo ir dydžio. Dažniausiai, jei kontrastingumas padidėja, tai dydis (skaidrės taškų skaičius) ženkliai išauga. Tai prailgina paveikslėlio apdorojimą, dėl to padidėja ir schemų sudėtingumas.

Didėjant sudėtingumui, mažėja praktinio pritaikymo galimybė. Skaidrių dydžiai tampa visiškai nevaldomi, ypač kai naudojama aukšta skiriamoji geba.

Bandant perduoti didelius kiekius informacijos, taip pat didėja ir sudėtingumas. Kol šifruojamos žodžiai ar frazės, schemos veikia efektyviai, tačiau, jei bandoma užšifruoti daugiau duomenų, pavyzdžiui, pastraipą, skaidrių dydžiai vėl tampa didžiuliai ir nevaldomi. Tai yra opi vizualiosios kriptografijos problema, nes dėl didelio skaidrių dydžio yra sunku jas panaudoti praktikoje.

2.2. Tikimybiniai vizualiosios kriptografijos algoritmai

Klasikiniame vizualiosios kriptografijos modelyje vienas taškas yra padalijamas į kelis potaškius. Tai vadinama taškų išplėtimu (angl. *Pixel expansion*). Taškų išplėtimas turi kelis trūkumus. Jis paveikia atkurto paveikslėlio kokybę ir visos schemos dydį. Apskritai, kokybę lemia taškų išplėtimas ir kontrastingumas. Kad spręsti taškų išplėtimo problemą, Yang [CY05] pasiūlė naują vizualiosios kriptografijos modelį, kuriame paveikslėlio atkūrimas yra tikimybinis, bet skaidrės išlieka tokio pat dydžio kaip ir paslaptis, t. y. schemose nėra išplečiami taškai. Pirmasis, kuris bandė kurti vizualiosios kriptografijos schemas be taškų išplėtimo buvo Ito [IKT99]. Abejuose Ito ir Yang modeliuose paslaptis atkuriama naudojant OR operaciją kiekvienam taškui skirtingose skaidrėse. Tokie modeliai yra vadinami tikimybiniais, nes nėra jokios garantijos, kad atkurtas taškas bus toks koks buvo: kai kuriais atvejais atkuriamas netoks taškas. Tai skiriasi nuo klasikinio modelio, kuriame baltas (juodas) taškas, gali būti atkuriamas, kaip pilkas. Kadangi tikimybiniuose modeliuose paslapties taškas yra teisingai atkuriamas su tam tikra tikimybe, tai atkurtos paslapties kokybė priklauso nuo tos tikimybės dydžio.

Įmanomas klasikinio ir tikimybinio modelių sujungimas. Galima kai kuriuos taškus padalinti, kad gautume didesnę tikimybę teisingai atkurti tašką. Ir galima nevisuose taškuose naudoti taškų išplėtimą, kad sumažintume schemos dydį.

Skaidres siūloma atvaizduoti n x m pasiskirstymo matrica M [CY12]. Kur kiekviena eilutė atvaizduoja vieną skaidrę, t. y. m potaškių, kur kiekvienas elementas yra 0 – baltam potaškiui ir 1 – juodam.

Tikimybinei schemai apibrėžiama tikimybė $p_{i|j}$, kuri nurodo tikimybę, kad buvo atkurtas taškas i, o paslėptame paveiksliuke buvo taškas j, kur $i, j \in \{b, w\}$.[3] Taigi, tikimybė $p_{w|w}(Q)$ nurodo tikimybę, kad buvo teisingai atkurtas baltas taškas sudėjus skaidres iš aibės Q. Kur Q yra baigtinė dalyvių aibė. O tikimybė $p_{b|w}(Q)$, nurodo tikimybę, kad buvo neteisingai atkurtas baltas taškas.

Skirtumai

$$p_{b|b}(Q) - p_{b|w}(Q)$$

ir

$$p_{w|w}(Q) - p_{w|b}(Q)$$

[3] b – juoda (angl. black); w – balta (angl. white)

įvertina schemos gerumą aibėje Q. Kuo skirtumai didesni, tuo geresnė schema. Jei abu dydžiai yra 1, visoms baigtinėms aibėms, tai turime klasikinį variantą, kur paslaptis visada bus atkuriama teisingai.

Jei egzistuoja teigiama konstanta β bent vienai baigtinei aibei Q

$$p_{b|b}(Q) - p_{b|w}(Q) \geq \beta$$

ir

$$p_{w|w}(Q) - p_{w|b}(Q) \geq \beta$$

tada schema yra vadinama β-tikimybinė, tai reiškia, kad β pažymi galimą klaidą atkūrime.

$$P_{w|w}(Q) = 1 - p_{b|w}(Q)$$

ir

$$p_{b|b}(Q) = 1 - p_{w|b}(Q)$$

todėl

$$p_{w|w}(Q) - p_{w|b}(Q) = p_{b|b}(Q) - p_{b|w}(Q) \quad (1)$$

Kadangi l ir h yra slenksčiai, kurie nurodo reikiamą potaškių skaičių, kad atskirti baltą tašką nuo juodo gali atsirasti tokių schemų ir tokių baigtinių aibių Q, kuriose juodų potaškių skaičius bus tarp l ir h, dėl to (1) lygybė nebus teisinga. Todėl siūloma nustatyti reikšmę $l = h - 1$ tokiu atveju (1) lygybė visada bus teisinga.

Pavyzdys 1. [CY12] Sakykime turime klasikinę 3 iš 4 schemą, kurios parametrai yra m = 6, h = 5, l = 4. Turėsime balto ir juodo taško bazines matricas:

$$M_w = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} M_b = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Konstruojame tikimybinę schemą S. Kiekviena matricos eilutė atvaizduoja vienos iš keturių skaidrių. Kiekvienas paslapties taškas plečiamas iki šešių ir šifruojamas pagal matricą, priklausomai nuo jo spalvos (Jei taškas baltas šifruojama pagal M_w, jei taškas juodas pagal M_b). Bandant atidengti imsime tris eilutes (nes schema yra 3 iš 4) ir atliksime OR operaciją stulpeliams. Taigi tikimybes gauname tokias $p_{b|b}(S)$ = 5/6, $p_{w|b}(S)$ = 1/6, $p_{w|w}(S)$ = 1/3, $p_{b|w}(S)$= 2/3, β = γ(S) = 1/6.

2.3. Vizualioji kriptografija ir atsitiktinės gardelės

Kafri ir Keren [KK87] 1997 metais apibrėžė atsitiktinę gardelę, kaip permatomumą apimantį dvimatį taškų masyvą. Kiekvienas taškas yra arba visiškai permatomas, arba matinis ir pasirinkimas tarp šių alternatyvų vykdomas monetos metimu. Taigi, nėra jokio sąryšio tarp skirtingų taškų reikšmių masyve.

Galime šifruoti dvejetainius paveikslėlius ar figūras į dvi atsitiktines gardeles, tokias, kad tik informaciją saugančios vietos abiejose gardelės siejasi tarpusavyje, tuo tarpu, visos kitos reikšmės yra visiškai atsitiktinės. Kai dvi gardelės yra sudedamos viena ant kitos, dėl šviesos pralaidumo (angl. *lighttransmission*) informaciją saugančios vietos išsiskirs iš atsitiktinio fono, taip pasimatys užšifruotas paveikslėlis ar figūra.

Dvejetainis taškas r yra vadinamas atsitiktiniu tašku, jei galimybė taškui tapti permatomu ar matiniu atsitiktinėje gardelėje R yra visiškai atsitiktinė. T. y. tikimybė, kad taškas r bus permatomas yra lygi tikimybei, kad taškas r bus matinis.

$$p(r = 0) = p(r = 1) = ½,$$

kur 0 žymi permatomą tašką, o 1 – matinį. Kadangi permatomas taškas praleidžia šviesą, o matinis ją sustabdo, tai atsitiktinio taško r vidutinis šviesos pralaidumas yra ½ ir yra žymima

$$t(r) = ½.$$

Kadangi atsitiktinės gardelės R kiekvienas taškas turi tą pačią tikimybę tapti permatomu, kaip ir matiniu, tai atsitiktinės gardelės R vidutinis šviesos pralaidumas irgi yra ½, bei žymimas

$$T(R) = ½.$$

OR operaciją žymėkime simboliu $\otimes$. Aišku, kad $r \otimes r$ ($R \otimes R$) yra visiškai tas pats kas tiesiog r (R), todėl

$$t(r) = ½ \text{ arba } T(R) = ½,$$

kiekvienam taškui r iš R.

Sakykime, kad R_1 ir R_2 yra dvi, tokio pat dydžio, nepriklausomos atsitiktinės gardelės. Kai jas uždedame vieną ant kitos, kiekvienas taškas (permatomas ar matinis) iš R_1 turi vienodą tikimybę būti uždengtas permatomo ar matinio taško iš R_2. $r_1 = R_1 [i, j]$ vadinsime $r_2 = R_2 [i', j']$ atliktimi, jei $i = i'$ ir $j = j'$ (jei r_1 pozicija R_1 gardelėje sutampa su r_2 pozicija R_2 gardelėje). Nesunku pastebėti, kad gardelių perdengimo tvarka yra nesvarbi, t. y.:

$$R_1 \otimes R_2 = R_2 \otimes R_1$$

Sudėjus du atitinkamus taškus r_1 ir r_2 tik viena iš keturių galimų kombinacijų bus permatoma (**Lentelė 2**). Kadangi visos keturios kombinacijos gali įvykti su tokia pačia tikimybe, tai tikimybė, kad $r_1 \otimes r_2$ bus permatomas yra ¼. Tai reiškia, kad vidutinis šviesos pralaidumas sudėjus R_1 ir R_2 (r_1 ir r_2) yra ¼.

r_1	r_2	$r_1 \otimes r_2$
0	0	0
0	1	1
1	0	1
1	1	1

Lentelė 2 Dviejų atsitiktinių taškų perdengimas

3. Vizualiosios kriptografijos algoritmų su atsitiktinėmis gardelėmis tyrimas

3.1. Atsitiktinių gardelių metodas dvispalvių vaizdų šifravimui

Pirmiausia buvo pasirinkti du algoritmai. Dvejetainio paveikslėlio dalijimas į dvi atsitiktines gardeles.[CY12]

Įvestis. w x h dydžio dvejetainis paveikslėlis B, kur $B[i, j] \in \{0, 1\}$ (baltas arba juodas), $1 \leq i \leq w$ ir $1 \leq j \leq h$.

Išvestis. Dvi atsitiktinės gardelės (skaidrės) R_1 ir R_2, kurias sudėjus vieną ant kitos pasimato slaptas paveikslėlis B, kur $R_k[i, j] \in \{0, 1\}$ (permatomas arba matinis), $1 \leq i \leq w$, $1 \leq j \leq h$ ir $k \in \{1, 2\}$.

Algoritmas 1.

Generuoti R_1 kaip atsitiktinę gardelę, $T(R_1) = ½$

// **for** (kiekvienam taškui iš $R_1[i, j]$, $1 \leq i \leq w$ ir $1 \leq j \leq h$) **do**

// R_1[i, j] = atsitiktinis_taškas(0, 1)

for (kiekvienam taškui $B[i, j]$, $1 \leq i \leq w$ ir $1 \leq j \leq h$) **do**

{ **if** ($B[i, j] = 0$) **then** $R_2[i, j] = R_1[i, j]$

else R_2[i, j] = atsitiktinis_taškas(0, 1)

}

Išvesti(R_1, R_2)

Algoritmas 2.

Generuoti R_1 kaip atsitiktinę gardelę, $T(R_1) = ½$

for (kiekvienam taškui $B[i, j]$, $1 \leq i \leq w$ ir $1 \leq j \leq h$) **do**

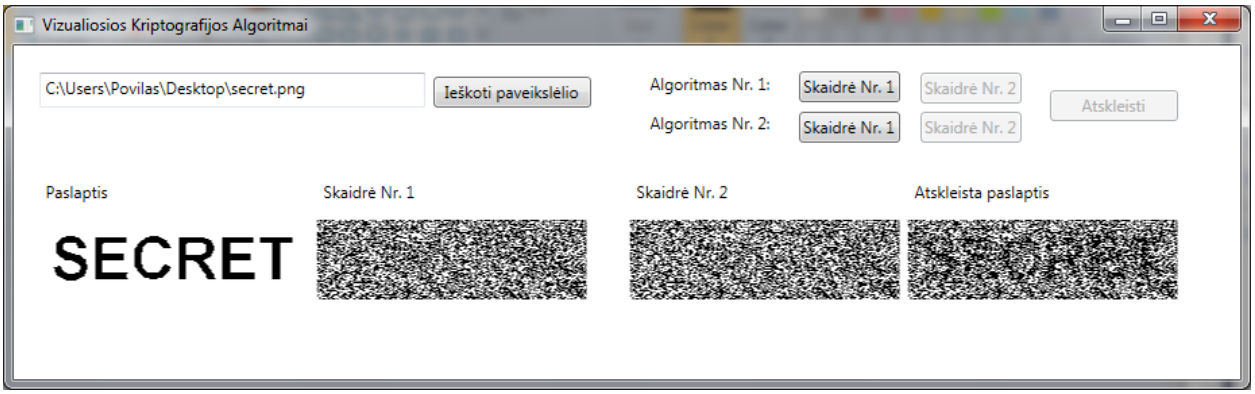
{ **if** ($B[i, j] = 0$) **then** $R_2[i, j] = R_1[i, j]$

else $R_2[i, j] = \overline{R_1[\imath, \jmath]}$

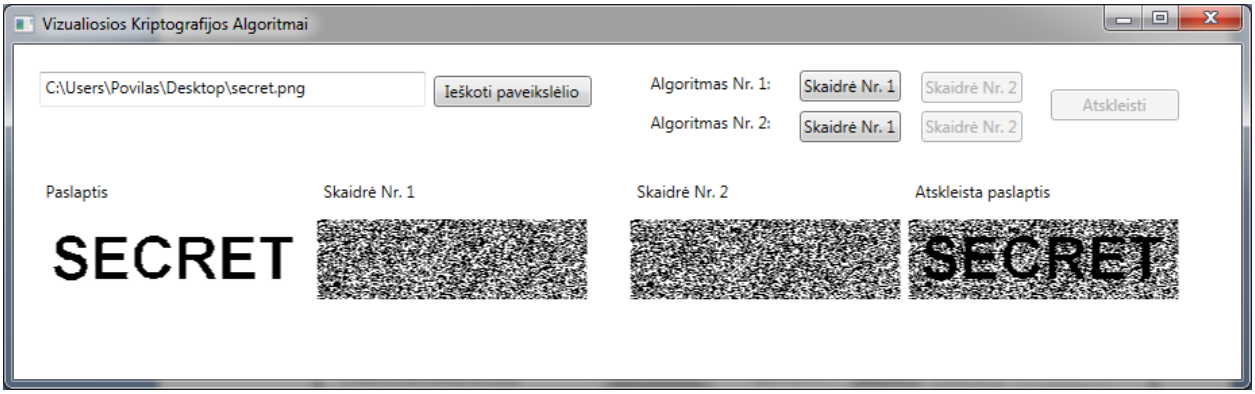
}

Išvesti(R_1, R_2)

Šie algoritmai buvo realizuoti programiškai ir gauti rezultatai:



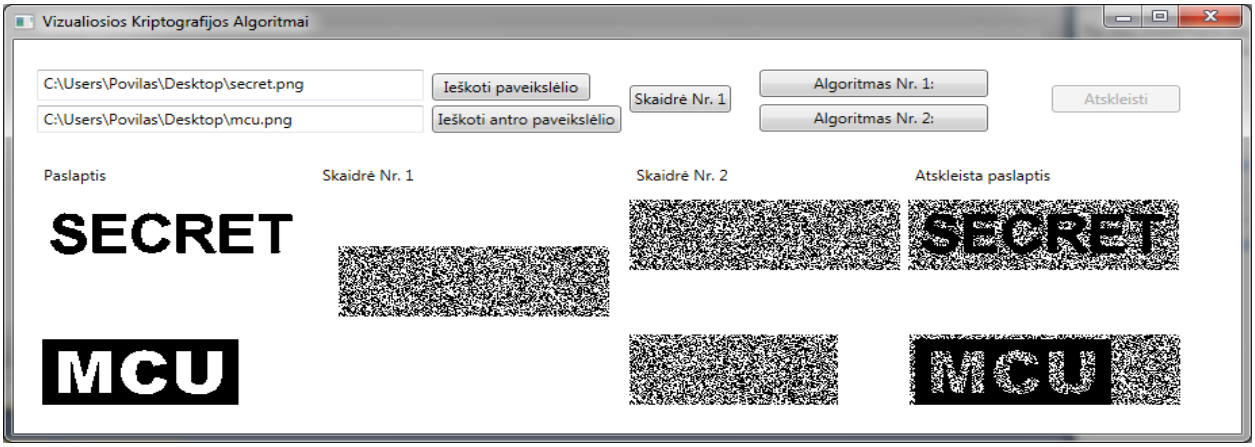
Pav. 5 Pirmojo algoritmo rezultatas



Pav. 6 Antrojo algoritmo rezultatas

Pirmasis algoritmas visus paslapties taškus atkūrė 51 – 52% tikslumu, o antrasis 55 – 56% tikslumu. Į šiuos skaičiavimus įtraukiama ir balta spalva, kuri šiuo atveju yra naudojama kaip fonas. Pati paslaptis, šiuo atveju yra žodis, t. y. juodi taškai, juos pirmasis algoritmas atkūrė 74-75% tikslumo, o antrasis 100% tikslumu. Kaip ir buvo tikėtasi antrasis algoritmas geriau atkūrė paslaptį, nes pirmasis algoritmas parenka antrosios skaidrės taškus atsitiktinai. Jei paslaptyje taškas yra juodas, tokiu atveju dažnai antrosios skaidrės taškas būna baltas (permatomas). Kadangi pirmoji skaidrė sugeneruota atsitiktinai atsiranda galimybė, kad abiejų skaidrių taškai bus balti, todėl jas sudėjus rezultate taškas taip pat bus baltas, nors paslaptyje jis yra juodas. Antrajame algoritme, tas taškas kuris paslaptyje buvo juodas, kuriant antrąją skaidrę bus šifruojamas atvirkščiai nei jis yra užšifruotas pirmojoje skaidrėje, todėl rezultate taškas niekad nebus baltas.

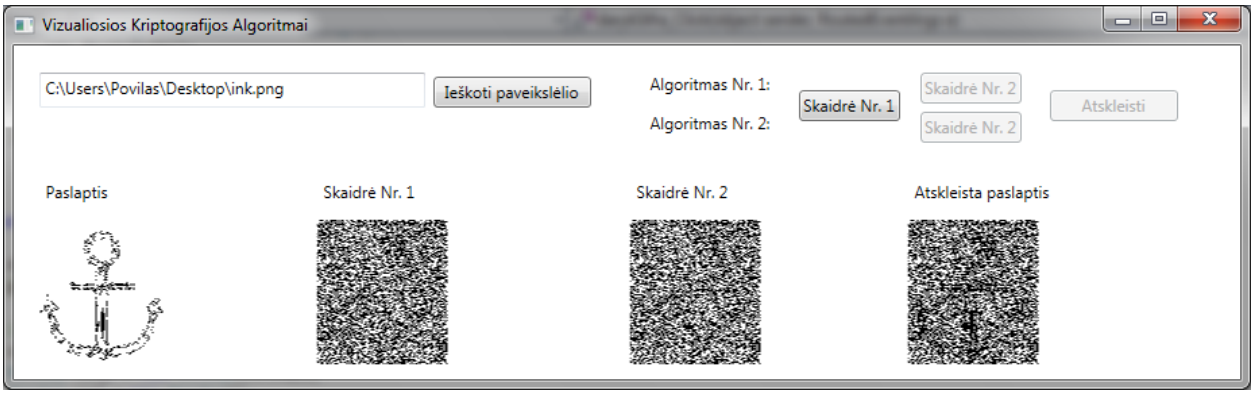
Kadangi algoritmas pirmąją skaidrę generuoja visiškai atsitiktinai ir nepriklausomai nuo paslapties, su ta pačia skaidre galima atkurti kelias skirtingas paslaptis.



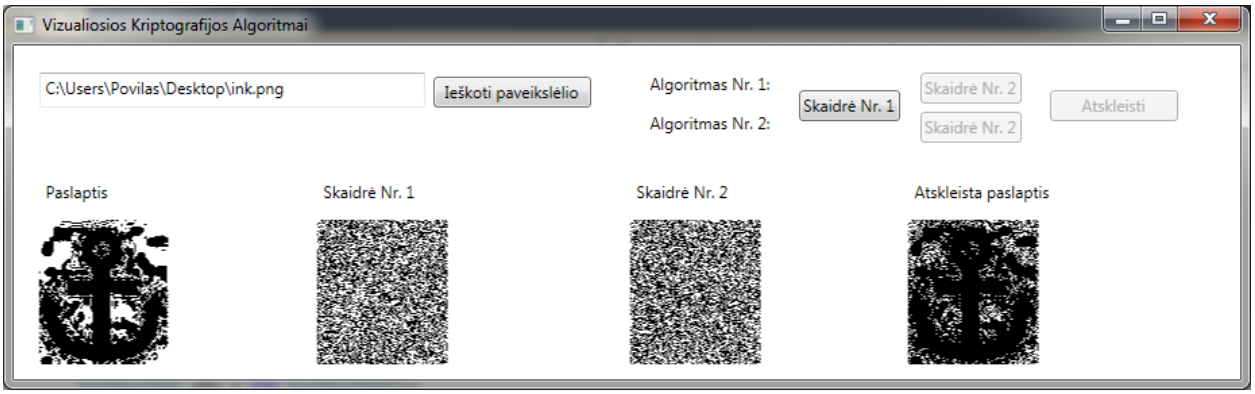
Pav. 7 Dvi paslaptys atkurtos su ta pačia skaidre

3.2. Tonavimo algoritmai

Dvispalvių vaizdų šifravimo algoritmai nesusidoroja su nespalvotais (*angl.* grayscale) paveikslėliais. Kadangi spalvos nebe dvi, o yra daug pilkos atspalvių. Visus pilkos spalvos atspalvius prijungus prie vienos ar kitos taškų aibės paveikslėliai yra išdarkomi.



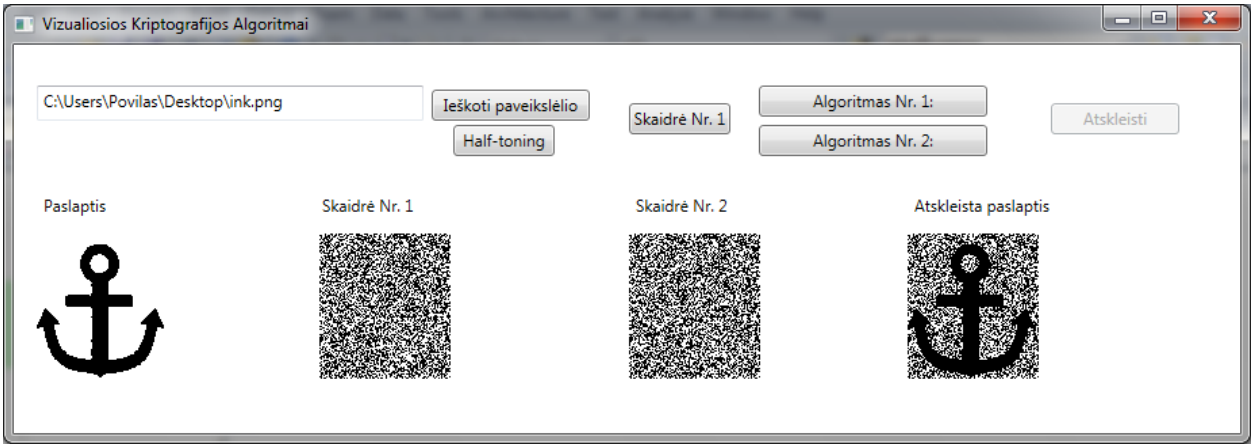
Pav. 8 Nespalvotas paveikslėlis pilkus atspalvius nuspalvinus baltai



Pav. 9 Nespalvotas paveikslėlis pilkus atspalvius nuspalvinus juodai

Ši problema buvo išspręsta naudojant tonavimo (*angl.* halftone) algoritmus. Kiekviena spalva susideda yra užkoduota RGB sistema[4] [Poy03]. Šioje sistemoje naudojamos trys, žmogaus akių receptorius atitinkančios spalvos: raudona (Red), žalia (Green) ir Mėlyna (Blue). RGB sistemoje spalva nagrinėjama, kaip spinduliavimas ir yra žymima trijų skaičių sąrašu arba kitaip – vektoriumi. Jei visų paveikslėlio taškų trys skaičiai esantys vektoriuje yra vienodi, tai paveikslėlis yra nespalvotas. Nulinės reikšmės atitinka juodą spalvą, o maksimalios (255) reikšmės – baltą. Reikšmės nuo 1 iki 254 atitinka pilkos spalvos atspalvius. Grubiausias algoritmas dalina reikšmes į dvi aibes beveik per vidurį, t. y. visos reikšmės mažesnės už 127 bus juodos, o didesnės bus baltos. Mūsų atveju atspalviai yra labai arti juodos ar baltos spalvos, todėl šis algoritmas su uždaviniu susidoroja puikiai.

[4] Apie RGB sistemą plačiau aprašyta kitame skyriuje



Pav. 10 Nespalvotas paveikslėlis pritaikius tonavimo algoritmą

Toks grubus algoritmas puikiai veikia su paveikslėliais, kurie ir taip yra panašūs į dvejetainius. Jei norėtume užšifruoti paveikslėlį, kuris turi daug pilkų atspalvių, reiktų naudoti sudėtingesnius algoritmus, pavyzdžiui Floyd–Steinberg drebinimo (*angl.* dither) algoritmą su klaidų skaičiavimu [CY12]. Drebinimo algoritmai skirsto pilkus taškus į juodus ir baltus taip, kad žmogaus akiai jie atrodytų pilki. Kad pasijaustų šis efektas, reikia šiuos veiksmus atlikti su didesnės rezoliucijos paveikslėliais, tam kad būtų daugiau taškų.

Įvestis. w x h dydžio nespalvotas paveikslėlis B, kur $B[i, j] \in \{0, 1, \ldots, 255\}$ (tarp balto ir juodo), $1 \le i \le w$ ir $1 \le j \le h$.

Išvestis. w x h dydžio dvejetainis paveikslėlis B, kur $B[i, j] \in \{0, 1\}$ (baltas arba juodas), $1 \le i \le w$ ir $1 \le j \le h$.

Floyd–Steinberg algoritmas

for (kiekvienam taškui $B[i, j]$, $1 \le i \le w$ ir $1 \le j \le h$) **do**
  senas_taškas = taškas[i, j]
  naujas_taškas = *artimiausia_spalva*(senas_taškas)
  taškas[i, j] = naujas_taškas
  kvant_klaida = senas_taškas – naujas_taškas
  taškas[i+1, j] = taškas[i+1, j] + kvant_klaida * 7/16
  taškas[i-1, j+1] = taškas[i-1, j+1] + kvant_klaida * 3/16
  taškas[i, j+1] = taškas[i, j+1] + kvant_klaida * 5/16
  taškas[i+1, j+1] = taškas[i+1, j+1] + kvant_klaida * 1/16
Išvesti(B)

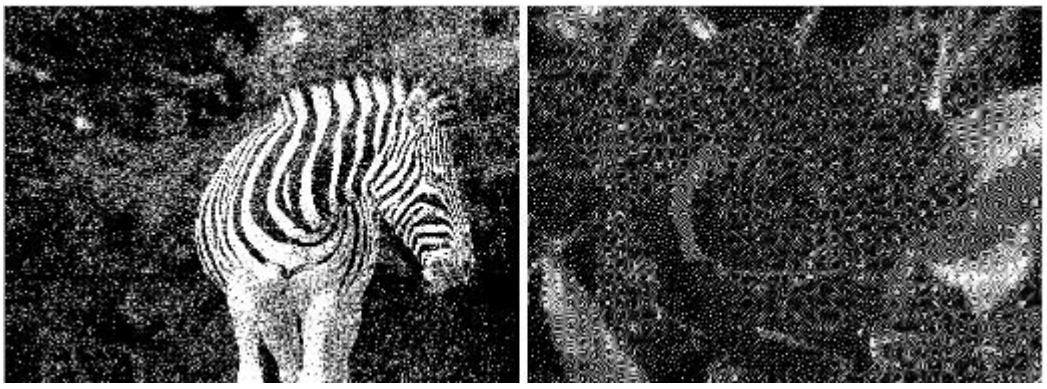
Floyd–Steinberg algoritmas pirmiausia priskiria taškui artimiausios spalvos reikšmę, paskaičiuoja atstumą tarp seno taško ir naujo ir padidina arba sumažina aplinkinių taškų reikšmes

tuo atstumu su tam tikra tikimybe. Kadangi taškai yra tikrinami iš kairės į dešinę ir iš viršaus į apačią, tai keičiamos tik dar netikrintų aplinkinių taškų reikšmės. Tikimybių ir taškų išsidėstymas parodytas **Lentelėje 3**. Taip keičiantis aplinkinių taškų reikšmėms, jiems turės būti priskirta kitokia reikšmė. Taigi, jei vienas taškas buvo priskirtas mažesnę reikšmę turinčiai spalvai, kitas turi daugiau šansų būti priskirtas didesnę reikšmę turinčiai spalvai. Jei paveikslėlyje yra keli gretimi taškai, kurių reikšmė yra tiesiai per vidurį tarp baltos ir juodos spalvų reikšmių, tai šie taškai, kitaip tariant ši dėmė, bus nuspalvinta kaip šachmatų lenta.

	Dabartinis taškas	7/16
3/16	5/16	1/16

Lentelė 3 Tikimybių ir taškų išsidėstymas

Kaip jau minėjau, šiam algoritmui labai svarbus paveikslėlio dydis, t. y. taškų kiekis. Tai galime matyti iš **pav. 11**, kur pavaizduoti šio algoritmo rezultatai skirtingo dydžio paveikslėliams: dideliam (1901 x 1069) ir mažam (475 x 357).



Pav. 11 Floyd–Steinberg tonavimo algoritmo rezultatai skirtingų dydžių paveikslėliams

Iš rezultatų lengva atskirti, kad paveikslėlis su zebru buvo didelis, o paveikslėlis su rože mažas. **Pav. 12** parodo kaip originaliai atrodė paveikslėliai.

Pav. 12 Paveikslėliai prieš tonavimą

3.3. Vizualioji spalvotų vaizdų kriptografija

3.3.1. RGB modelis

Pagrindinė ypatybė konstruojant vizualiosios kriptografijos schemas nespalvotiems paveikslėliams yra tokia: jei sudedame skaidres pilnas juodų ir baltų taškų, tai rezultate gautas taškas bus juodas, jei bent vienas iš sudėtų taškų buvo juodas, o gautas taškas bus baltas tik tuo atveju, jei visi sudėti taškai buvo balti. Kitaip sakant, bandant atskleisti paslaptį, žmogaus akis atlieka OR operaciją.

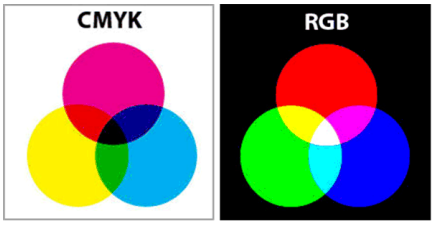
Dirbant su spalvotais vaizdais, ši ypatybė nėra taip lengvai pritaikoma, nes bandant atskleisti paslaptį mūsų akys naudoja sudėtingesnę operaciją, nei paprastą OR. Kad suprasti kas vyksta bandant atskleisti paslaptį, reikia prisiminti fiziką, šviesos ir spalvų teoriją.

Šviesos ir spalvų supratimas yra svarbus dalykas dirbant su vizualiąja spalvotų vaizdų kriptografija. Grubiai tariant, šviesa susideda iš elektromagnetinio spinduliavimo, kurio bangų ilgis yra tarp 350 - 750 *nm*. Matoma šviesa yra tik maža dalis viso elektromagnetinio spektro dalis. Kai tam tikro ilgio banga pasiekia žmogaus akies tinklainę, ji yra suprantama kaip spalva. Matomos šviesos spektre, trumpesnio ilgio bangos suprantamos, kaip melsva spalva, vidutinio ilgio bangos, kaip žalsvos spalvos, o ilgiausios bangos kaip rausvos spalvos. Jei akį pasiekia visos bangos iš matomos šviesos spektro, tai jos suprantamos kaip balta spalva.

Objektas yra tam tikros spalvos, todėl kad kai šviesa į jį atsimuša, jis kai kurias elektromagnetines bangas sugeria, o kai kurias atspindi. Tam tikros spalvos *x* objektas sugeria visas bangas kurios nėra spalvos *x* ir atspindi tos spalvos bangas. Pavyzdžiui, objektas atrodo geltonas, nes jis atspindi geltoną šviesą, o sugeria visas kitas šviesos spektro bangas. Permatomo objekto atveju, šviesa kuri į jį nesusigeria, vietoj to, kad atsispindėtų, pereina kiaurai objektą.

Spalvų modeliai leidžia atvaizduoti visas spalvas. Dažniausiai naudojamas spalvų modelis yra vadinamas priemaišinis spalvų modelis (*angl.* additive color model). Pagrindinėmis spalvomis pasirinkus raudoną, žalią ir mėlyną, galime išgauti daug gamos spalvų. Toks modelis vadinamas RGB. Jis labai paplitęs ekranuose (kompiuterių, televizorių, telefonų ir t.t.).

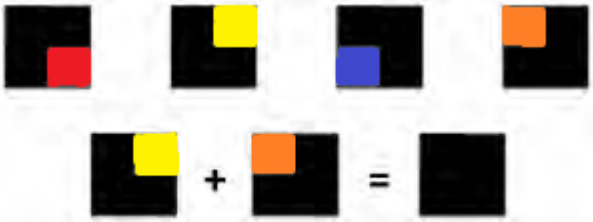
Kitas spalvų modelis, atvirkštinis RGB modeliui yra CMY modelis. Jis sugeria spalvas, dažnai naudojamas CMYK modelis, pridedant juodą spalvą.



Pav. 12 CMYK ir RGB spalvų modeliai

3.3.2. Algoritmas spalvotoms paslaptims

Dauguma egzistuojančių vizualiosios spalvotų vaizdų kriptografijos modelių, naudoja tos pačios spalvos (*angl*. Same color (SC)) algoritmus. Jų esmė neleisti dviem skirtingoms spalvoms persidengti. Tam yra įvesta naikinimo spalva – juoda. Verheul ir van Tilborg[VT97] buvo vizualios spalvotų vaizdų kriptografijos pradininkai. Jų pasiūlytas algoritmas neleidžia skirtingų spalvų taškams persidengti. Veikimo principas yra toks: naudojama klasikinė *k* iš *n* schema, taškas yra padalijamas į *c* dalių, kur *c* yra paslapties skirtingų spalvų skaičius. O taškų praplėtimas yra c^k. Potaškis (*angl.* subpixel) *i* gauna spalvą numeriu *i*. o likę potaškiai nuspalvinami juodai (**pav. 13** [CY12]).



Pav. 13 Algoritmas 4 spalvų atveju. Skirtingos spalvos niekada nepersidengia.

Šis algoritmas nėra tinkamas paveikslėliams šifruoti, nes beveik viskas būtų užšifruota juodai. Sakykime turime 3 iš 3 schemą, užšifruotą trimis spalvomis. Taškai išsiplėstų iki $3^3 = 27$. ir gautume 26 iš 27 juodus taškus (apie 96%). Nors šis algoritmas paveikslėlių šifravimui nėra praktiškas, tačiau jis gali būti panaudojamas kitokiems dalykams, pavyzdžiui, slaptažodžių perdavimui, kai vienas skaičius yra viena spalva. Jei naudotume 0,5 *cm* diametro taškus su 9 spalvoms, galime sugeneruoti 3 iš 9 vizualią schemą su devyniomis spalvomis. Naudojant $9^2 = 81$ tašką kiekvienai spalvai. Ant paprasto A4 lapo tilptų 90 skaitmenų slaptažodis.

Bendrasis modelis, kur skirtingos spalvos perdengia viena kitą yra mažiausiai ištyrinėtas literatūroje, pateiktos tik kelios schemos, todėl buvo pasirinkta paeksperimentuoti šioje srityje. Pasitelkus RGB spalvų modelį, buvo pabandyta šifruoti trijų spalvų paveikslėlius. Trečioji spalva gaunama sujungus dvi spalvas iš RGB modelio.

Visi toliau aprašyti algoritmai spalvotiems vaizdams buvo kuriami tobulinant pirmuosius du algoritmus dvispalviams vaizdams. Pasiūlytuose algoritmuose, kuriant skaidres vietoj dviejų spalvų (juodos ir baltos) naudotos atitinkamai trys, keturios ir septynios skirtingos spalvos.

3.3.3. Pirmasis algoritmas trijų spalvų paslaptims

Pirmasis siūlomas algoritmas yra su taškų praplėtimu, vienas taškas buvo praplėstas iki devynių taškų. Skaidrėms naudojamos trys spalvos: permatoma, raudona ir žalia.

Įvestis. w x h dydžio trijų spalvų paveikslėlis B, kur $B[i, j] \in \{0, 1, 2\}$ (geltonas, raudonas arba žalias), $0 \leq i < w$ ir $0 \leq j < h$.

Išvestis. Dvi atsitiktinės gardelės (skaidrės) R_1 ir R_2, kurias sudėjus vieną ant kitos pasimato slaptas paveikslėlis B, kur $R_k[i, j] \in \{0, 1, 2\}$ (permatomas, raudonas arba žalias), $0 \leq i < w*3$, $0 \leq j < h*3$ ir $k \in \{1, 2\}$.

Algoritmas 3.

Generuoti R_1 kaip atsitiktinę gardelę, $T(R_1) = ⅓$

// **for** (kiekvienam taškui iš $R_1[i, j]$, $0 \leq i < w*3$ ir $0 \leq j < h*3$) **do**

// R_1[i, j] = atsitiktinis_taškas(0, 1, 2)

for (kiekvienam taškui $B[i, j]$, $0 \leq i < w$ ir $0 \leq j < h$) **do**

for (kievienam taškui $R_1[x,y]$, $i*3 \leq x < i*3+3$ ir $j*3 \leq y < j*3+3$) **do**

switch (spalva)

{ **case** ($B[i, j] = 0$)

if ($R_1[x,y] = 0$) **then** $R_2[x, y] = 0$

else if ($R_1[x,y] = 1$) then $R_2[x, y] = 2$

else $R_2[x,y] = 1$

case ($B[i, j] = 1$)

if ($R_1[x,y] = 0$) **then** $R_2[x, y] = 1$

else if ($R_1[x,y] = 1$) then $R_2[x, y] = 0$

else $R_2[x,y] = 2$

case ($B[i, j] = 2$)

if ($R_1[x,y] = 0$) **then** $R_2[x, y] = 2$

else if ($R_1[x,y] = 1$) **then** $R_2[x, y] = 1$

else $R_2[x,y] = 0$

}

Išvesti(R_1, R_2)

Algoritmo principas yra toks: kuriant antrąją skaidrę vienas paslapties taškas yra lyginamas su devyniais pirmosios skaidrės taškais. **Lentelėje 4** parodyti galimi variantai.

Paslapties taško spalva	Pirmosios skaidrės taško spalva	Antrosios skaidrės taško spalva
Geltona	Permatoma	Permatoma
Geltona	Raudona	Žalia
Geltona	Žalia	Raudona
Raudona	Permatoma	Raudona
Raudona	Raudona	Permatoma
Raudona	Žalia	Žalia
Žalia	Permatoma	Žalia
Žalia	Raudona	Raudona
Žalia	Žalia	Permatoma

Lentelė 4 Galimi šifravimo variantai

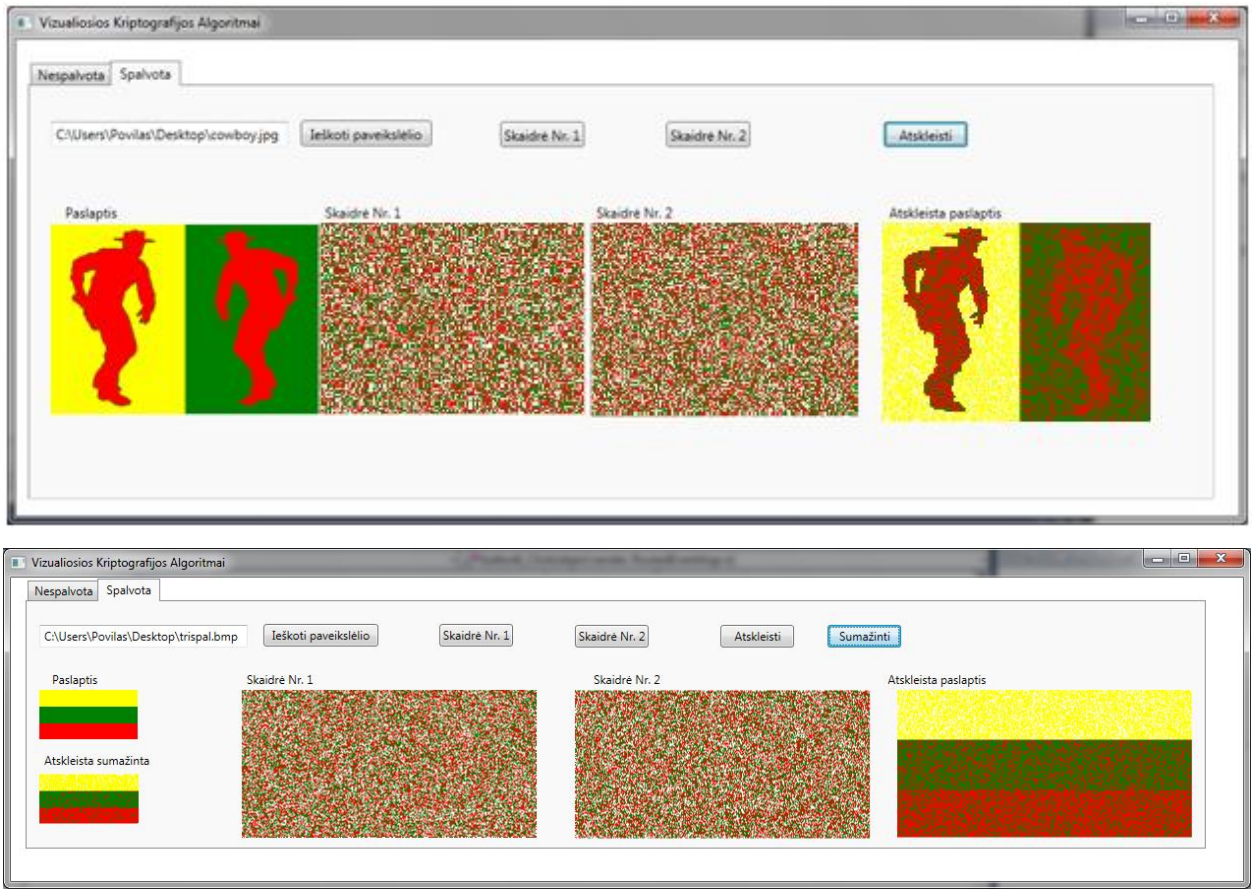
Iš pradžių atsitiktinė spalva buvo priskiriama tada, kai originalios spalvos nebegalime išgelbėti, pavyzdžiui, paslapties taškas buvo raudonas, o atsitiktinai sugeneravus pirmąją skaidrę taškas gavosi žalias. Kad ir ką priskirtume antrojoje skaidrėje, atskleidus paslaptį taškas bus arba žalias arba geltonas, raudono gauti neįmanoma. Tačiau, tokiu atveju, antroje skaidrėje matydavosi paslapties kontūrai, todėl algoritmas buvo pakeistas, taip, kad atsitiktinio spalvos rinkimo nebūtų ir kiekviena paslapties spalva šifruojama vienodu kiekiu spalvų.

Norint atskleisti paslaptį, kitaip tariant dešifruoti, reikia sujungti du toje pačioje vietoje esančius taškus, skaidres sudėti vieną ant kitos. **Lentelėje 5** parodyti galimi variantai.

Pirmosios skaidrės taško spalva	Antrosios skaidrės taško spalva	Rezultatas
Permatoma	Permatoma	Permatoma
Permatoma	Raudona	Raudona
Permatoma	Žalia	Žalia
Raudona	Permatoma	Raudona
Raudona	Raudona	Raudona
Raudona	Žalia	Geltona
Žalia	Permatoma	Žalia
Žalia	Raudona	Geltona
Žalia	Žalia	Žalia

Lentelė 5 Paslapties atskleidimo variantai

Pabaigoje, buvo sukurta sumažinimo funkcija, kad būtų galima aiškiau matyti paslapties ir atskleistos paslapties skirtumus, kai jos yra vienodo dydžio. Funkcijos principas yra iš devynių, vieną paslapties tašką žyminčių taškų išrinkti dominuojančią spalvą (t. y. spalvą, kuria yra nudažyta dauguma iš tų devynių taškų) ir ja nuspalvinti naują tašką. Praktiniai rezultatai pavaizduoti **Pav. 14**.



Pav. 14 Praktiniai rezultatai

Kadangi dirbama su kompiuterio pagalba, o ne sudėtas skaidres laikome prieš šviesą, galima pagudrauti ir šiek tiek paryškinti geltoną spalvą permatomus taškus keičiant geltonais. Juk vienintelis atvejis, kai pirmoje ir antroje skaidrėje abu taškai yra permatomi, bus tada, kai paslapties paveikslėlyje taškas buvo geltonas.

Pav. 15 Rezultatai pakeitus permatomus taškus geltonais

Galima matyti, kad sumažintos paslapties kokybė šiek tiek pagerėjo. Prieš šį pakeitimą, algoritmas paslapties taškus atkurdavo 85% tikslumu, o dabar atkuria 90% tikslumu.

3.3.4. Antrasis algoritmas trijų spalvų paslaptims

Antrasis siūlomas algoritmas buvo konstruojamas pagal idėją [CY12] naudoti taškų naikinimą, bei taško praplėtimą, vienas taškas buvo praplėstas iki keturių taškų. Paslaptis yra trijų skirtingų spalvų, o skaidrės keturių spalvų: raudona, žalia, permatoma ir juoda.

Įvestis. w x h dydžio trijų spalvų paveikslėlis B, kur $B[i, j] \in \{0, 1, 2\}$ (geltonas, raudonas arba žalias), $0 \leq i < w$ ir $0 \leq j < h$, bei spalvų aibė $S[i] \in \{0, 1, 2, 3\}$ (permatomas, raudonas, žalias, arba juodas).

Išvestis. Dvi skaidrės R_1 ir R_2, kurias sudėjus vieną ant kitos pasimato slaptas paveikslėlis B, kur $R_k[i, j] \in \{0, 1, 2, 3\}$ (permatomas, raudonas, žalias, arba juodas), $0 \leq i < w*2$, $0 \leq j < h*2$ ir $k \in \{1, 2\}$.

Algoritmas 4.

for (kiekvienam taškui $B[i, j]$, $0 \leq i < w$ ir $0 \leq j < h$) **do**
$S[] = \{0, 1, 2, 3\}$
index = atsitiktinis_skaičius(S elementų skaičius) // atsitiktinė spalva iš spalvų aibės
for (kiekvienam praplėstiems taškams x, y, $i*2 <= x < i*2+2$ ir $j*2 <= y < j*2+2$) **do**
{
    $R_1[x,y] = S$[index]
    switch (spalva)
    {
    case ($B[i, j] = 0$)
      if (S[index] = 0) **then** $R_2[x, y] = 3$
        else if (S[index] = 1) **then** $R_2[x, y] = 2$
          else if (S[index] = 2) **then** $R_2[x, y] = 1$
            else $R_2[x,y] = 0$
    case ($B[i, j] = 1$)
      if (S[index] = 0) **then** $R_2[x, y] = 1$
        else if (S[index] = 1) **then** $R_2[x, y] = 0$
          else if (S[index] = 2) **then** $R_2[x, y] = 3$
            else $R_2[x,y] = 2$
    case ($B[i, j] = 2$)

if (S[index] = 0) **then** $R_2[x, y] = 2$

else if (S[index] = 1) **then** $R_2[x, y] = 3$

else if (S[index] = 2) **then** $R_2[x, y] = 0$

else $R_2[x,y] = 1$

}

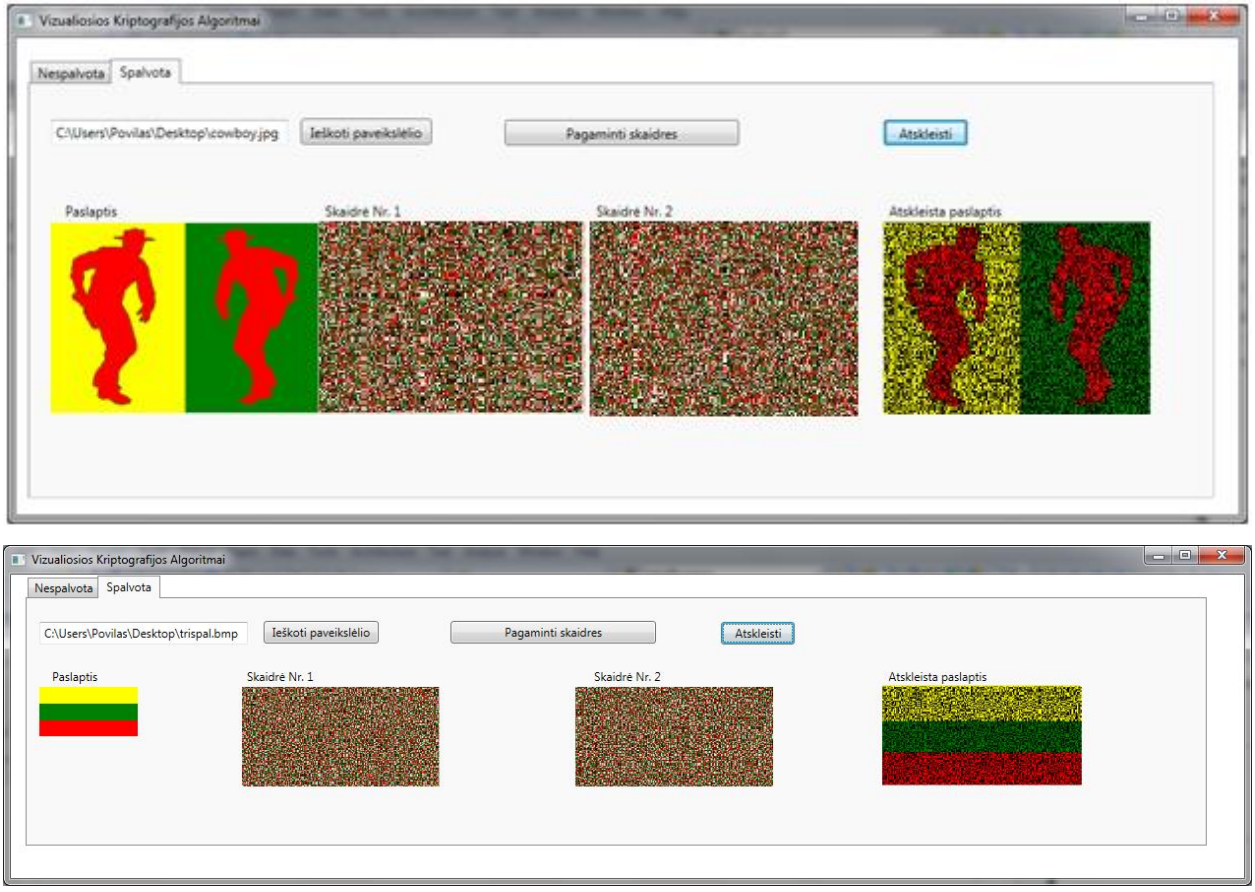
$S = S \setminus \{S[\text{index}]\}$

index = atsitiktinis_skaičius(S elementų skaičius)

}

Išvesti(R_1, R_2)

Šis algoritmas, skirtingai nuo prieš tai buvusių, pirmąją skaidrę kuria neatsitiktinai, o vieną jos tašką praplečia į tiek taškų, kiek yra spalvų, šiuo atveju į keturis taškus. Ir visus juos nuspalvina skirtingomis spalvomis. Antrąją skaidrę kuria pagal pirmąją, naudodamas juodą spalvą naikina nereikalingas spalvas, todėl atskleidus paslaptį visada pusė praplėsto taško bus juoda, pusė reikiamos spalvos. Taigi algoritmo tikslumas yra 50%, tačiau, žinant tokią informaciją galime juodus taškus nuspalvinti ta pačia spalva kokia nuspalvinti šalia jų esantys taškai ir paslaptis bus atkurta 100% tikslumu.



Pav. 16 Algoritmo Nr. 4 rezultatai

Kaip matome pasitelkus spalvų maišymo savybes, galima skaidrėse neturėti tam tikros spalvos, tačiau ją užšifruoti ir sėkmingai atskleisti. Daugelis literatūroje aprašytų algoritmų neišnaudoja šių savybių.

3.3.5. Algoritmas šešių spalvų paslaptims

Prieš tai naudoti algoritmai apdorojo paveikslėlius sudarytus iš trijų spalvų, o skaidrėse buvo panaudoti skaidrūs arba juodi taškai. Sekančiam algoritmui buvo pasitelktas pilnas RGB spalvų modelis. Todėl paslapties paveikslėlis jau gali būti šešių spalvų, o skaidrės sudarytos iš keturių spalvų: raudonos, žalios, mėlynos ir baltos (skaidrus taškas). Algoritmas sudarytas sujungus algoritmą Nr. 3 ir algoritmą Nr. 4. Vienas paslapties taškas yra praplečiamas iki keturių taškų, o pirma ir antra skaidrės yra kuriamos vienu metu: pirmoji atsitiktinai, o antroji – lyginant pirmąją su paslaptimi.

Įvestis. w x h dydžio šešių spalvų paveikslėlis B, kur $B[i, j] \in \{0, 1, 2, 3, 4, 5\}$ (raudonas, žalias, mėlynas, geltonas, žydras, purpurinis), $0 \leq i < w$ ir $0 \leq j < h$.

Išvestis. Dvi atsitiktinės gardelės (skaidrės) R_1 ir R_2, kurias sudėjus vieną ant kitos pasimato slaptas paveikslėlis B, kur $R_k[i, j] \in \{0, 1, 2, 3, 4, 5, 6\}$ (raudonas, žalias, mėlynas, geltonas, žydras, purpurinis, permatomas), $0 \leq i < w*2$, $0 \leq j < h*2$ ir $k \in \{1, 2\}$.

Lentelėje 6 parodyti galimi variantai kaip bus kuriama antroji skaidrė.

Paslapties taško spalva	Pirmosios skaidrės taško spalva	Antrosios skaidrės taško spalva
Raudona	Permatoma	Raudona
Raudona	Raudona	Permatoma
Raudona	Žalia	Žalia
Raudona	Mėlyna	Mėlyna
Žalia	Permatoma	Žalia
Žalia	Raudona	Raudona
Žalia	Žalia	Permatoma
Žalia	Mėlyna	Mėlyna
Mėlyna	Permatoma	Mėlyna
Mėlyna	Raudona	Raudona
Mėlyna	Žalia	Žalia
Mėlyna	Mėlyna	Permatoma
Geltona	Permatoma	Permatoma
Geltona	Raudona	Žalia
Geltona	Žalia	Raudona
Geltona	Mėlyna	Mėlyna
Žydra	Permatoma	Permatoma
Žydra	Raudona	Raudona
Žydra	Žalia	Mėlyna
Žydra	Mėlyna	Žalia

Purpurinė	Permatoma	Permatoma
Purpurinė	Raudona	Mėlyna
Purpurinė	Žalia	Žalia
Purpurinė	Mėlyna	Raudona

Lentelė 6 Algoritmo šešioms spalvoms šifravimo variantai

Kadangi algoritmas kurdamas skaidres praplečia kiekvieną paslapties tašką į keturis taškus ir tuos taškus nuspalvina keturiomis skirtingomis spalvomis galima matyti, kad sudėjus dvi skaidres visada du iš keturių taškų bus atskleisti teisingai. Jei pritaikytume mažinimo funkciją, kurią naudojau Algoritme Nr. 3 kiekviename keturių taškų rinkinyje dominuojanti spalva būtų teisinga ir algoritmo tikslumas būtų 100% procentų, kaip ir Algoritme Nr. 4 juodus taškus nuspalvinus ta pačia spalva kokia nuspalvinti šalia jų esantys taškai. Paveikslėlyje Nr. 17 pateikti praktiniai šio algoritmo rezultatai.

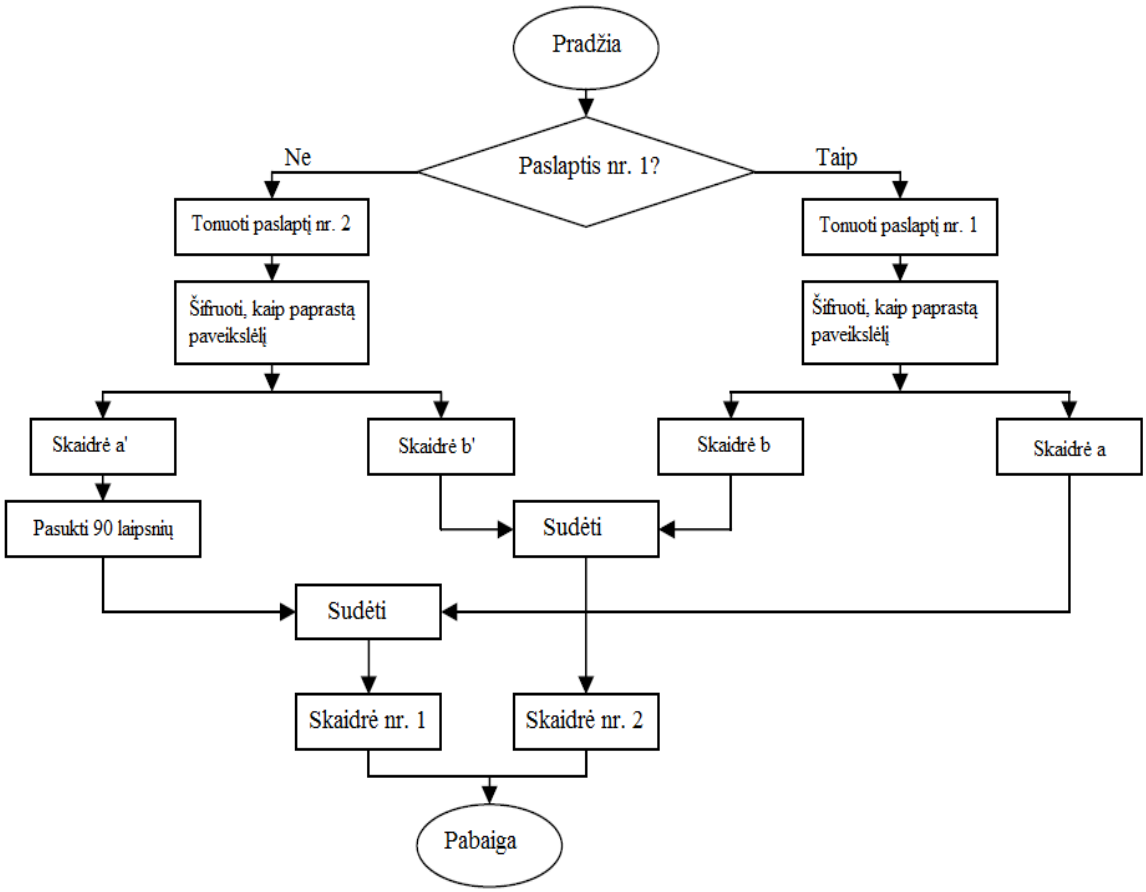
Pav. 17 Praktiniai rezultatai

Pereiti nuo nespalvotų paveikslėlių kriptografijos prie spalvotų nėra paprasta. Atsiranda spalvų maišymo problemos. Daugelis vizualiosios spalvotų vaizdų kriptografijos algoritmų šią problemą bando aplenkti tiesiog nemaišydami spalvų. Tik keletas algoritmų naudoja spalvų maišymą. Atsiranda saugumo, kontrastingumo problemos. Vizualiosios kriptografijos schemų konstravimas spalvotiems vaizdams yra sudėtingesnė ir mažai ištirta vizualiosios kriptografijos sritis.

3.4. Vizualioji kriptografija kelioms paslaptims

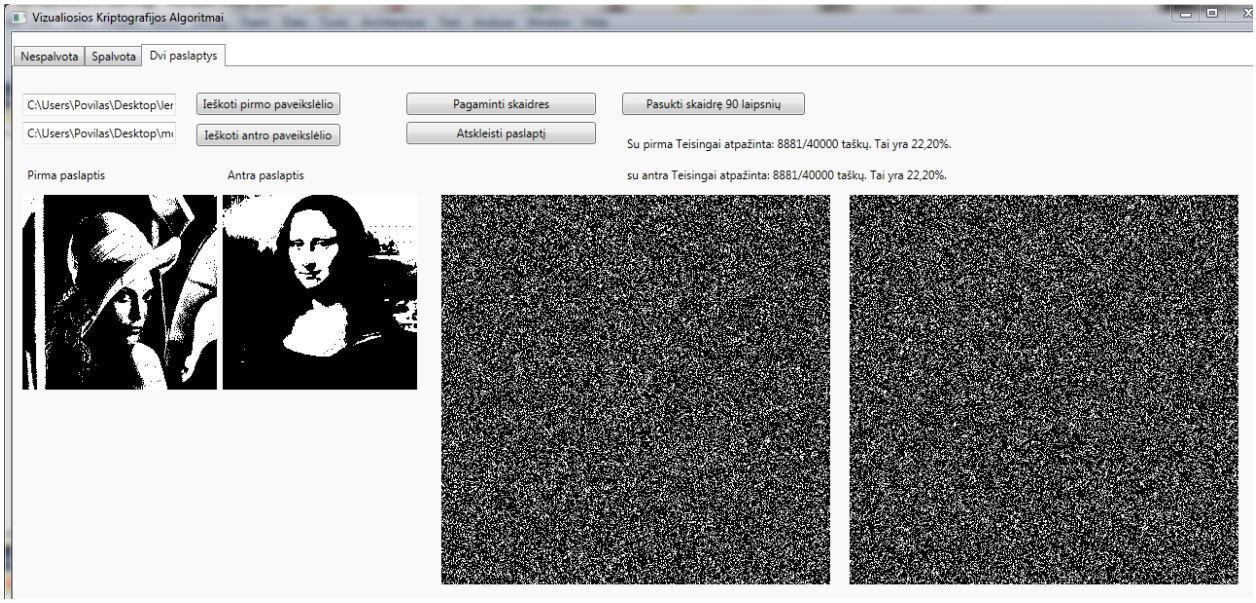
Norint užšifruoti kelias paslaptis, nebūtina kiekvienai paslapčiai kurti kitos skaidrės. Galima skaidres pagaminti taip, kad sudėjus vieną ant kitos pasimatytų pirmoji paslaptis, o vieną skaidrę pasukus 90^0 kampu, pasimatytų antroji. Kad tai pavyktų, paslaptys turi būti vienodo dydžio ir būti kvadrato formos.

Buvo pasirinktas algoritmas [DS11], kuris paslapties paveikslėlius pirmiausiai užšifruoja paprastai, tada atitinkamas skaidres sujungęs pagamina dvi galutines skaidres. Algoritmas naudoja taškų praplėtimą, taigi, turint du *x***x* dydžio paslapties paveikslėlius, gaminame dvigubai didesnes 2*x**2*x* skaidres. Pirmiausia, paslapties paveikslėliai yra užšifruojami atskirai, klasikiniu algoritmu t.y. iš pirmosios paslapties, gauname skaidres „a“ ir „b“, o iš antrosios – skaidrę „a`“ ir skaidrę „b`“. Sudėję skaidres „b“ ir „b`“ vieną ant kitos gauname antrąją skaidrę, o sudėję skaidrę „a“ ir $90^0$ laipsnių prieš laikrodžio rodyklę paverstą skaidrę „a`“ gauname pirmąją skaidrę. Pirmąją paslaptį atskleidžiame sudėję gautas skaidres vieną ant kitos, o antrąją gauname sudėję pirmąją skaidrę pasuktą 90^0 laipsnių pagal laikrodžio rodyklę su antrąja skaidre. Algoritmo veikimas pavaizduotas **diagramoje 1**.

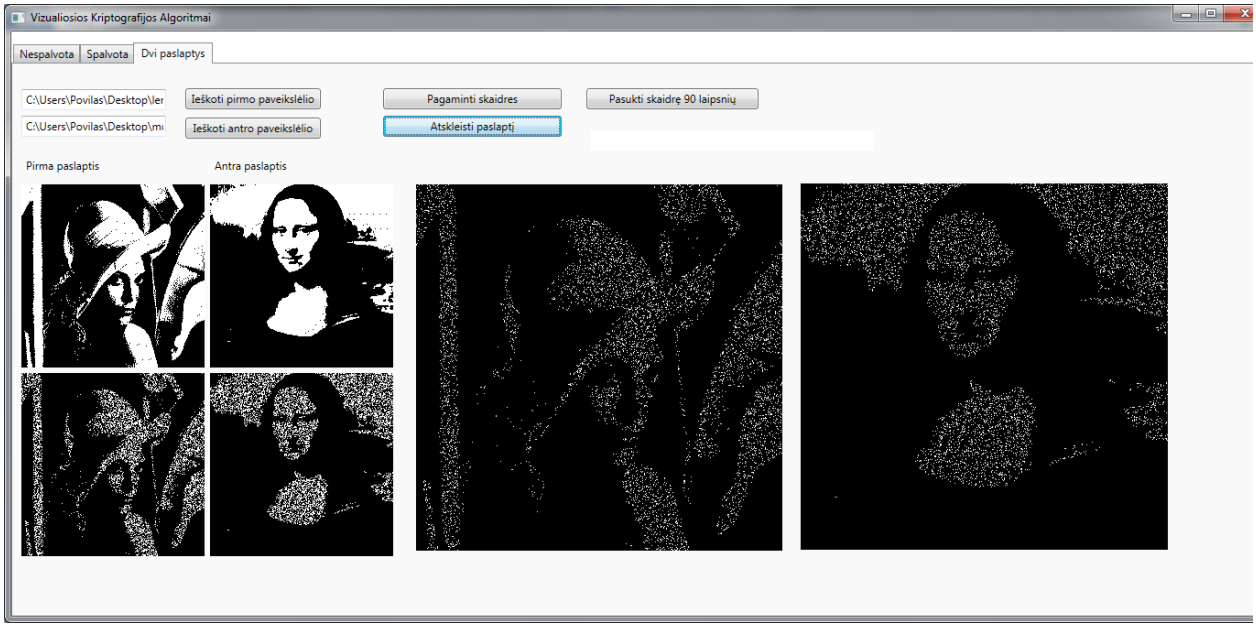


Diag. 1 Algoritmo veikimas

Šiame algoritme, kaip ir kituose algoritmuose su taškų praplėtimu, naudojau mažinimo funkciją. Mano naudotu pavyzdžiu, paslaptys paveikslėliuose buvo baltos spalvos, paslapčių dydis buvo 200x200, t. y. 40000 taškų. Algoritmas teisingai atkurdavo, apie 28000 taškų, t. y. 71-72%. Tačiau, jei skaičiuosime tik baltus taškus, juos atkurdavo tik 8% tikslumu, todėl teko pakeisti mažinimo funkciją. Kaip minėjau anksčiau, ši funkcija iš praplėstų taškų, pagamina vieną tašką ir jį nuspalvina, tokia spalva, kurios buvo daugiausiai tuose praplėstuose taškuose. Šiuo atveju iš keturių taškų dažnai trys būna juodi, nes pasukus skaidrę visada atsiranda daugiau juodų taškų. Vadinasi, mažinimo funkciją reiktų pakeisti, taip, kad jei bent vienas taškas yra baltas, sumažintas taškas turi būti baltas. Atlikus šį pakeitimą, rezultatai iškart pagerėjo, paslapties taškų atkūrimas nuo 8% pakilo iki 41%, o šie bendri taškai, pakėlė ir viso paveikslėlio atkūrimo tikslumą dešimčia procentų, iki 81-82% tikslumo.



Pav. 18 Užšifruotos skaidrės



Pav. 19 Algoritmo rezultatai

Paveikslėlyje 18 pavaizduotos dvi užšifruotos skaidrės, o **paveikslėlyje 19** du rezultatai jas sudėjus. Pirmasis sudėjus dvi skaidres, o antrasis vieną skaidrę pavertus 90^0 laipsnių kampu ir skaidres sudėjus.

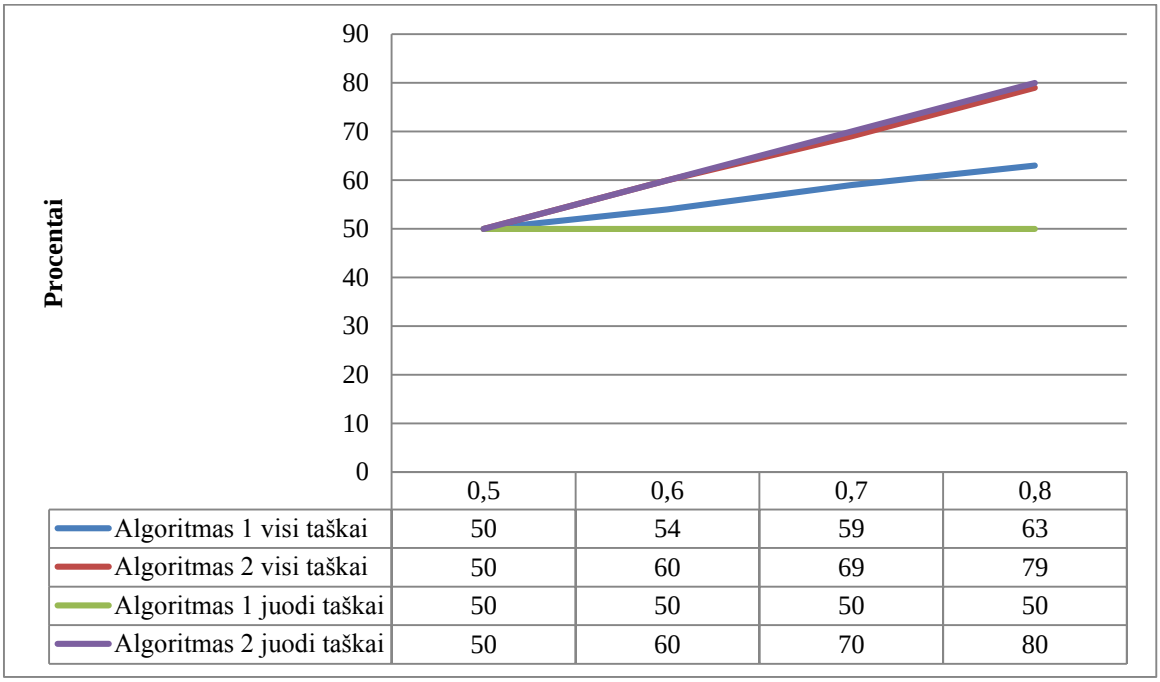
Bandant šifruoti kelias paslaptis nukenčia baltos spalvos atkūrimas, todėl šis algoritmas labiau tinkamas šifruoti paslaptis, kuriose svarbi informacija yra šifruojama juoda spalva, arba gautą atsakymą mažinti kompiuterio pagalba, taip ryškinant kontrastingumą.

4. Šifravimo saugumo tyrimas

4.1. Algoritmai dvispalviams paveikslėliams

Šifro saugumas priklauso nuo skaidrių parengimo. Kadangi antroji skaidrė parengiama pagal pirmąją, tai viskas priklauso nuo to, kaip parenkami pirmos skaidrės taškai. Jeigu kiekvienas taškas su tikimybe 1/2 parenkamas kaip juodas arba baltas, tai šifravimas saugus. Jeigu balti taškai parenkami su didesne už 1/2 tikimybe, tai rengiant antrąją skaidrę daugiau jos juodų taškų sutampa su paveikslo juodais taškais.

Buvo atliktas tyrimas su keturiomis tikimybėmis: 0,5, 0,6, 0,7, 0,8. Paslapties paveikslėlis su antrąja skaidre yra lyginamas dviem būdais: kiek procentų taškų sutampa su visu paveikslėliu ir kiek procentų juodų taškų sutampa su paslapties juodais taškais. Paslapties paveikslėlio išvaizda nėra svarbi. T. y. ar paslaptyje kaip fonas naudojama balta ar juoda spalvos neturi įtakos rezultatams. Rezultatai pavaizduoti **Grafike 1**.



	0,5	0,6	0,7	0,8
Algoritmas 1 visi taškai	50	54	59	63
Algoritmas 2 visi taškai	50	60	69	79
Algoritmas 1 juodi taškai	50	50	50	50
Algoritmas 2 juodi taškai	50	60	70	80

Grafikas 1 Tyrimo rezultatai

Iš rezultatų galime matyti, kad pirmasis algoritmas atskleidžia mažiau paslapties taškų nei antrasis. Tai įvyksta dėl tos pačios priežasties, dėl kurios pirmasis algoritmas blogiau atkuria paslaptį pabaigoje, antrosios skaidrės taškus jis dažnai kuria atsitiktinai, tuo tarpu antrasis algoritmas tuos taškus parenka atvirkščius pirmajai skaidrei, todėl nenuostabu, kad didinant baltų taškų kiekį pirmojoje skaidrėje didės juodų taškų skaičius antrojoje.

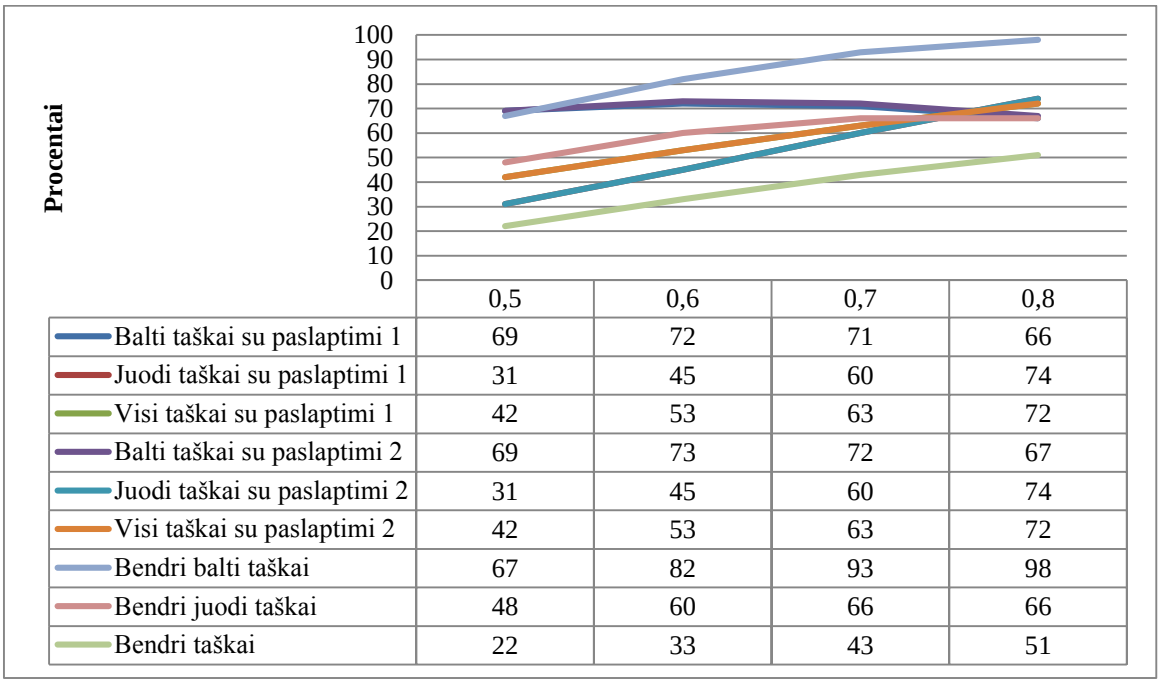
Taigi, pirmasis algoritmas yra saugesnis, tačiau jei visi taškai sutampa bent 60% paslaptis tampa pakankamai matoma, bei dėl didesnio baltų taškų skaičiau abejose skaidrėse nukenčia atkurtos paslapties kokybė. Kaip matome antrasis algoritmas tampa nesaugus jau prie tikimybės

0,6, o pirmasis tik prie tikimybės 0,7. Juodų taškų lyginimo atveju, pirmasis algoritmas kuria ima antros skaidrės tašką atsitiktinai, jei paslaptyje taškas buvo juodas, todėl nenuostabu, kad pirmosios skaidrės keitimas įtakos neturėjo. Tačiau antrasis algoritmas tokiu atveju kuria tašką priešingos spalvos, nei buvo pirmojoje skaidrėje, todėl rezultatai sutapo su tikimybe.

Kitas tirtas algoritmas buvo algoritmas su keliomis paslaptimis. Čia buvo pasirinktos tos pačios keturios tikimybės. Gauti rezultatai:

- Kiek antrosios skaidrės baltų taškų sutampa su pirma paslaptimi.
- Kiek antrosios skaidrės juodų taškų sutampa su pirma paslaptimi.
- Kiek antrosios skaidrės taškų sutampa su pirma paslaptimi.
- Kiek antrosios skaidrės baltų taškų sutampa su antra paslaptimi.
- Kiek antrosios skaidrės juodų taškų sutampa su antra paslaptimi.
- Kiek antrosios skaidrės taškų sutampa su antra paslaptimi.
- Kiek antrosios skaidrės baltų taškų sutampa su paslapčių taškais, kurie abejuose paslaptyse balti.
- Kiek antrosios skaidrės juodų taškų sutampa su paslapčių taškais, kurie abejuose paslaptyse juodi.
- Kiek antrosios skaidrės taškų sutampa su paslapčių taškais, kurie abejuose paslaptyse tokios pat spalvos.

Rezultatai pavaizduoti **Grafike 2**.



	0,5	0,6	0,7	0,8
Balti taškai su paslaptimi 1	69	72	71	66
Juodi taškai su paslaptimi 1	31	45	60	74
Visi taškai su paslaptimi 1	42	53	63	72
Balti taškai su paslaptimi 2	69	73	72	67
Juodi taškai su paslaptimi 2	31	45	60	74
Visi taškai su paslaptimi 2	42	53	63	72
Bendri balti taškai	67	82	93	98
Bendri juodi taškai	48	60	66	66
Bendri taškai	22	33	43	51

Grafikas 2 Tyrimo rezultatai

Kadangi šifruojamos dvi paslaptys, antrojoje skaidrėje pasimato dviejų paslapčių mišinys, todėl negalima iškart pamatyti visos paslapties. Geriausiai matosi dalys, kuriose abiejų paslapčių taškai sutampa, todėl bandant šifruoti tą pačią paslaptį du kartus, ji visa tampa matoma antroje skaidrėje. Dėl padidinto baltų taškų kiekio atskleistos paslapties kokybė suprastėja, tačiau vėl ten kur taškai sutampa, paveikslėlis yra atkurtas geriau.

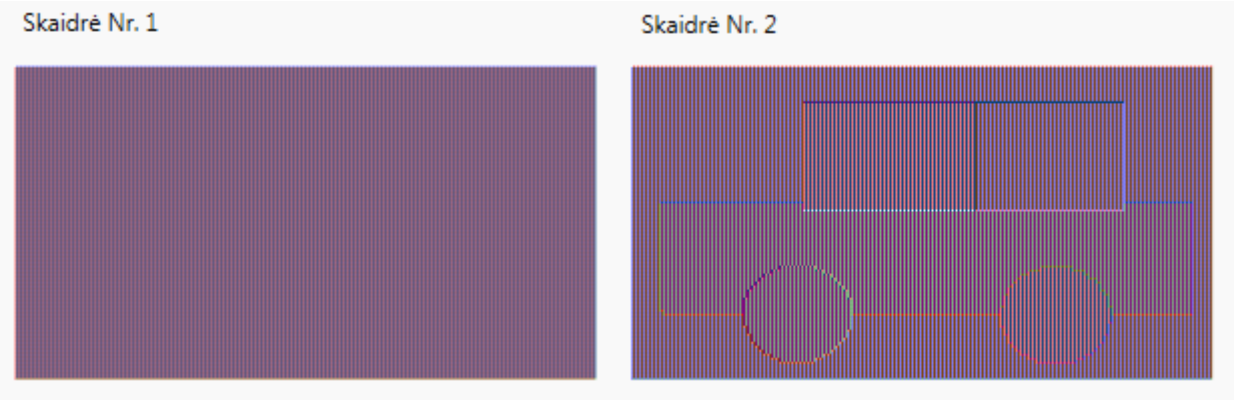
4.2. Algoritmas spalvotiems paveikslėliams

Nusprendžiau tyrimui naudoti algoritmą šešių spalvų paveikslėliams, nes jis yra patobulintas dviejų algoritmų trispalviams paveikslėliams variantas. Kaip ir tirtuose algoritmuose dvispalviams paveikslėliams, taip ir čia antroji skaidrė yra parengiama pagal pirmąją, todėl viskas priklauso nuo to, kaip parenkami pirmosios skaidrės taškai. Po taško išplėtimo yra 4! = 4 x 3 x 2 x 1 = 24 variantai kaip gali atrodyti keturi taškai, kurie atvaizduoja vieną paslapties tašką:

RS SR BG
BG, BG, ... , SR

Kur R – Raudona, B – Mėlyna, G – Žalia, S – Skaidri.

Svarbu, kad pirmoji skaidrė būtų ruošiama iš skirtingų variantų. Jei naudojame tik vieną iš variantų, tai antrojoje skaidrėje kiekviena spalva įgaus savo raštą ir taip bus galima atkurti paslaptį tik iš antrosios skaidrės.



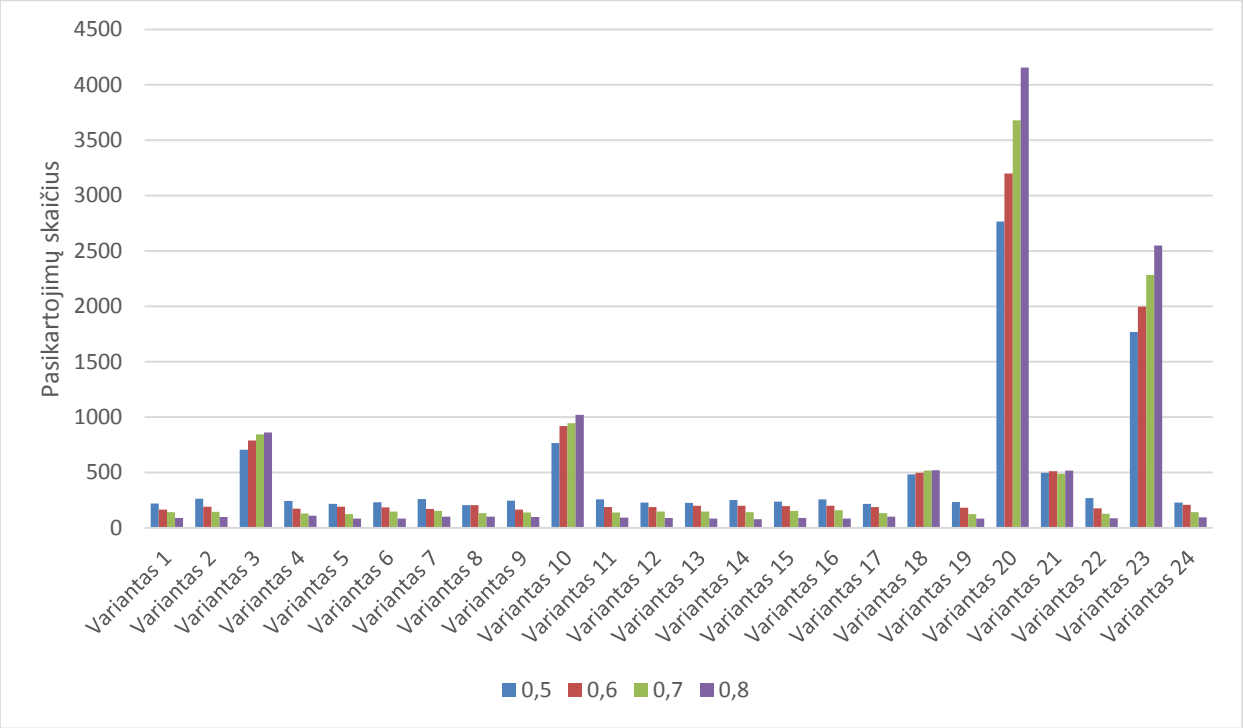
Pav. 20 Pirmosios skaidrės kūrimas naudojant vienintelį variantą

Žinant šią savybę buvo eksperimentuota kai vienas variantas pirmojoje skaidrėje pasitaiko dažniau už kitus. Tyrimas atliktas su keturiomis tikimybėmis: 0,5, 0,6, 0,7, 0,8. Tai reiškia, kad kuriant pirmąją skaidrę vienas iš dvidešimt keturių variantų buvo pasirenkamas atitinkamai 0,5, 0,6, 0,7 ir 0,8 kartų. Konkrečiu pavyzdžiu paslapties paveikslėlis sudarytas iš 145 x 78 = 11310 taškų. Praplėtus paveikslėlį variantų skaidrėje bus lygiai toks pat skaičius. Buvo suskaičiuota koks skaičius skirtingų variantų yra sugeneruojamas antrojoje skaidrėje ir kaip taip nulemia

vaizdą toje skaidrėje. Pvz. Pav. 19 antroji skaidrė buvo sugeneruota tik iš šešių skirtingų variantų. Rezultatai pateikti **Lentelėje 7** ir **Grafike 3**.

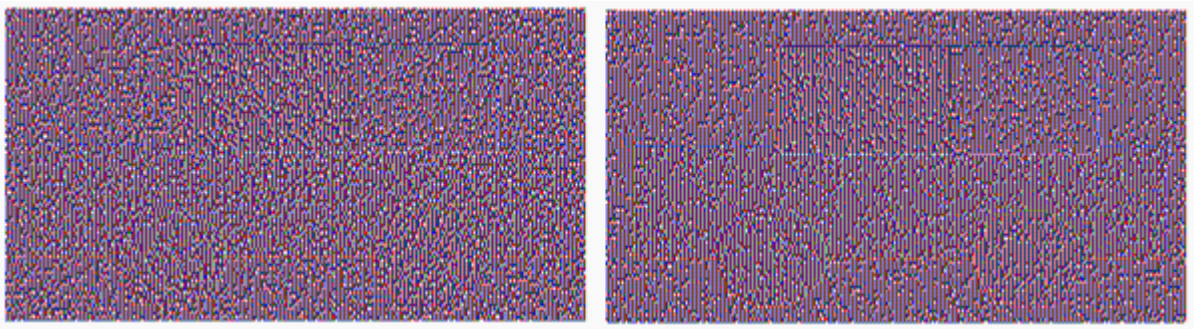
	0,5	0,6	0,7	0,8
Variantas 1	220	165	142	90
Variantas 2	263	192	146	99
Variantas 3	707	791	845	864
Variantas 4	244	176	130	112
Variantas 5	218	191	127	86
Variantas 6	232	186	149	85
Variantas 7	261	173	154	103
Variantas 8	208	206	134	102
Variantas 9	247	165	139	100
Variantas 10	768	919	945	1021
Variantas 11	259	189	141	93
Variantas 12	231	188	149	91
Variantas 13	228	201	148	84
Variantas 14	254	201	142	80
Variantas 15	237	198	154	92
Variantas 16	259	200	159	85
Variantas 17	217	189	134	102
Variantas 18	484	499	518	523
Variantas 19	236	184	127	86
Variantas 20	2767	3201	3681	4158
Variantas 21	498	512	490	520
Variantas 22	270	177	128	88
Variantas 23	1771	1997	2285	2550
Variantas 24	231	210	143	96

Lentelė 7 Variantų kiekis antrojoje skaidrėje



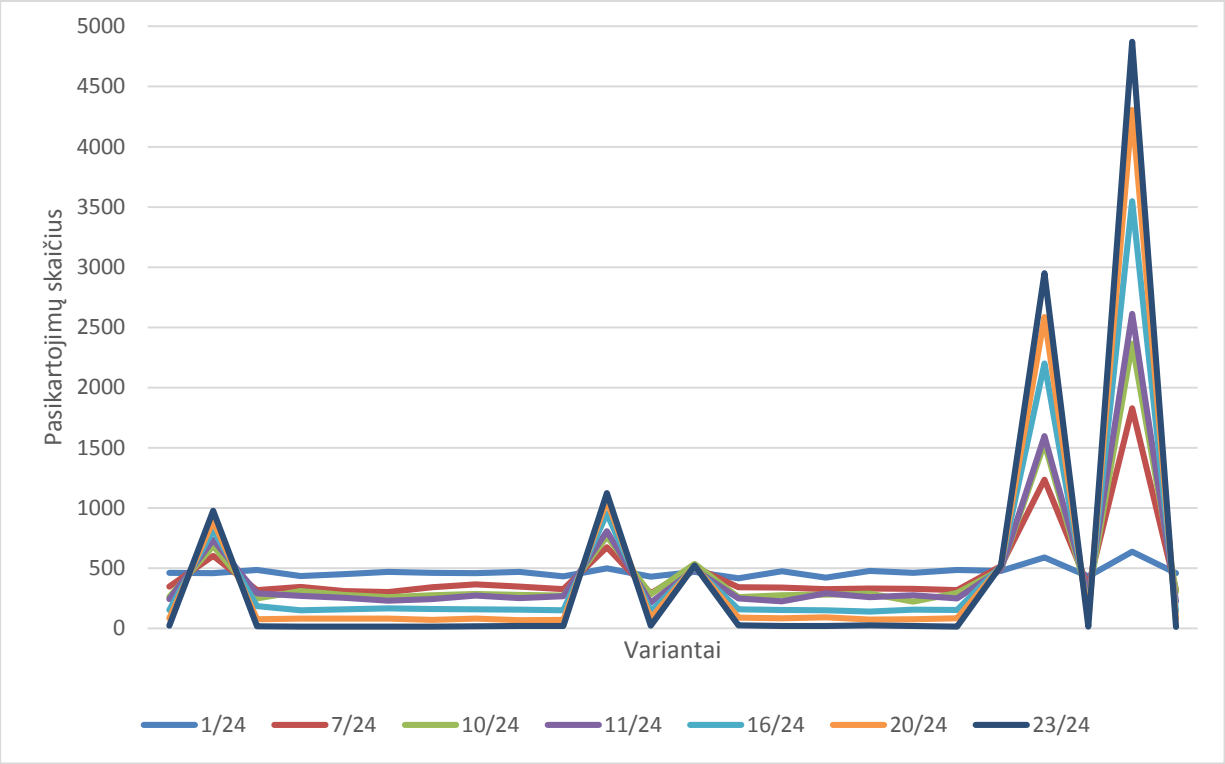
Grafikas 3 Variantų kiekis antrojoje skaidrėje

Pažvelgus į rezultatus matosi, kad keli variantai tiesiog dominuoja nepriklausomai nuo tikimybės, tačiau vaizdas skaidrėse labai skiriasi. Daugiau skirtingų variantų ir vaizdas tampa labiau margas, todėl žmogaus akiai iškart sunkiau pastebėti paslaptį. Pavyzdžiui palyginus atvejus kai skaidrė kuriama su tikimybe 0,5 ir su tikimybe 0,7, antruoju atveju keturių dominuojančių variantų kiekis antrojoje skaidrėje padaugėja apie 15%. O tai yra pakankamai daug ir tai iškart atsiliepia antros skaidrės vaizdui (**Pav. 21**).



Pav. 21 Antrųjų skaidrių vaizdas, kai pirmoji kuriama su tikimybe 0,5 (kairėje) ir su tikimybe 0,7 (dešinėje)

Toliau tyrimas buvo atliekamas pasirenkant dvidešimt tris skirtingas tikimybes nuo 1/24 iki 23/24. Su šiomis tikimybėmis buvo pasirinktas vienas iš dvidešimt keturių variantų. Po truputį didinat vieno varianto dažnumą buvo bandoma nustatyti iki kada algoritmas bus saugus. T. y. su kokia didžiausia tikimybe paslaptis vis dar nebus matoma. **Grafike 4** pavaizduoti skirtingų variantų skaičius kai vienas variantas yra naudojamas su viena iš tikimybių.



Grafikas 4 Skirtingų variantų skaičiai

Rezultate paslapties dalelė pradėjo ryškėti esant tikimybei 11/24. Šiuo atveju dažniausias variantas pasikartojo 2612 kartus iš 11310 esamų variantų. T.y. 23% skaidrės sudarė toks pats variantas. Galime daryti išvadą, kad jei skaidrėje vienas pasikartojantis variantas sudaro mažiau nei 23% visos skaidrės, šifravimas bus saugus.

Šiame algoritme pirmoji skaidrė yra naudojama kaip raktas, todėl keičiant ją paprasčiausia išdarkyti šifravimą ir taip bandyti atskleisti paslaptį iš antrosios skaidrės. Tačiau, net naudojant tokią aukštą tikimybę kaip 0,5 šifras lieka daugiau mažiau saugus, nes žmogaus akiai sunku įžiūrėti paslaptį, nors skirtingų variantų kiekiai labai skiriasi.

5. Išvados ir rezultatai

Darbe apžvelgta klasikinis ir tikimybinis vizualiosios kriptografijos metodai, supažindinta su atsitiktinėmis gardelėmis ir jų panaudojimu vizualiojoje kriptografijoje. Tirti algoritmai dvispalvių vaizdų šifravimui, buvo susidurta su tonavimo problema. Aptartas tonavimo algoritmas. Ištirtas algoritmas skirtas šifruoti dvi paslaptis vienu metu. Dvispalvių vaizdų šifravimui skirti algoritmai patobulinti ir pasiūlyti algoritmai trispalvių vaizdų šifravimui, pasitelkus RGB modelį algoritmai sujungti ir patobulinti iki šešių spalvų vaizdų šifravimo, aptartos spalvų maišymo problemos.

- Dvispalvių vaizdų vizualinės kriptografijos metodai pasižymi geru paslapties vaizdo kontūro atkūrimu, tačiau prarandamas kontrastingumas.

- Dvispalvių vaizdų algoritmai buvo lyginami ir tiriamas jų saugumas, didinant baltų taškų kiekį pirmojoje skaidrėje. Pirmasis algoritmas atskleidė mažiau paslapties taškų nei antrasis, tačiau jei visi taškai sutampa bent 60% paslaptis tampa pakankamai matoma.

- Algoritmas šifruoti dvi paslaptis vienu metu labiau tinkamas šifruoti paslaptis, kuriose svarbi informacija yra šifruojama juoda spalva, nes nukenčia kontrastingumas, bei dėl taškų išplėtimo padidėja skaidrių matmenys. Šiai problemai panaudota mažinimo funkcija, kuri didina kontrastingumą.

- Buvo tiriamas algoritmo šifruoti dvi paslaptis vienu metu saugumas. Pirmojoje skaidrėje buvo didinamas baltų taškų kiekis ir pastebėta, kad antrojoje skaidrėje pasimato dviejų paslapčių mišinys, todėl bandant šifruoti tą pačią paslaptį du kartus, ji visa tampa matoma antroje skaidrėje.

- Pasiūlyti trispalvių vaizdų šifravimo algoritmai pasižymi geru paslapties vaizdo kontūro atkūrimu, tačiau naikinant spalvas nukenčia tikslumas. Dėl taškų išplėtimo, sukurta mažinimo funkcija, kad su kompiuteriu gauti geresnį kontrastingumą.

- Tiriant algoritmo skirto šifruoti šešių spalvų paveikslėliams saugumą vienas taškas buvo išplėstas į keturis skirtingų spalvų taškus, ši aibė buvo pavadinta variantu ir ištirtas skaidrių saugumo santykis su pasikartojančių variantų dažnumu. Nors variantų dažnumas skyrėsi kelis kartus, tačiau iš antrosios skaidrės buvo beveik neįmanoma atskleisti paslapties. Toks netikėtas rezultatas, tik įrodė algoritmo saugumą ir dešifravimo žmogaus akimi galią.

- Buvo bandoma surasti su kokiu didžiausiu pasikartojančių variantų kiekiu algoritmas išliks saugus. Rasta, kad paslaptis skaidrėje pradeda ryškėti, kai jų kiekis yra 23%.

Vizualiosios kriptografijos algoritmai pasižymi išskirtiniu saugumu, todėl norint atskleisti paslaptį neturint visos informacijos galima tik keičiant pačių skaidrių kūrimo procesą. Dauguma sutinkamų algoritmų yra pritaikyti tik dvispalviams vaizdams, tačiau patobulinus šiuos algoritmus ir šifruojant paveikslėlius su daugiau spalvų jie tampa saugesni, nes žmogaus akis negali atskirti daug skirtingų susimaišiusių spalvų.

6. Šaltinių sąrašas

[WY12] J. Weir, W. Yan. *Visual Cryptography and Its Applications*. Ventus Publishing ApS, 2012. Prieiga per internetą: http://books.google.lt/books?id=iWioDoO_g90C&printsec=frontcover&dq=visual+cryptography

[NS95] M. Naor, A. Shamir. Visual cryptography. *Advances in Cryptology: EUROCRYPT 94*, 1995.

[CY12] S. Cimato, C. N. Yang. *Visual cryptography and secret image sharing*. CRC Press, Taylor& Francis Group, LLC, 2012. Prieiga per internetą: http://books.google.lt/books?id=WPzhnh0CI3MC&printsec=frontcover&dq=visual+cryptography+and+secret+image+sharing

[HKS00] T. Hofmeister, M. Krause, H. U. Simon. Contrast optimal k out of n Secret sharing schemes in visual cryptography. *Theoretical Computer Science*, 240(2):471-485, 2000.

[THLT05] P. Tuyls, H. D. L. Hollmann, J. H. Van Lint, L. Tolhuizen. XOR-based visual cryptography schemes. *Designs, Codesand Cryptography*, 37(1):169-186, 2005.

[Sha49] C. Shannon. Communication theory of secrecy systems. *Bell System Tchnical Journal*, 28(4), 656-715, 1949.

[Yan04] C. N. Yang. New visual secret sharing schemes using probabilistic method. *Pattern Recognition Letters*, 25, 481–494, 2004.

[YC05] C. N. Yang, T. S. Chen. Size-adjustable visual secret sharing schemes. *IEEE Trans. Fundamentals*, 88, 2471–2474, 2005.

[IKH99] R. Ito, H. Kuwakado, H. Tanaka. Image size invariant visual cryptography. *IEICE Transactionson Fundamentals of ElectronicsCommunicationsand Computer Sciences*, E82A(10), 2172-2177, 1999.

[KK87] O. Kafri and E. Keren. Encryption of pictures and shapes by random grids. *Opt. Lett*, 12:377-379, 1987.

[Dro96] S. Droste. New results on visual cryptography, Advances in Cryptology: CRYPTO96, 1109:401-415, 1996.

[VT97] E.R. Verheul, H.C.A. Van Tilborg. Constructions and properties of k out of n visual secret sharing schemes. *Designs Codes Cryptography*, 11:179-196, 1997.

[KS03] M. Krause, H. U. Simon. Determining the optimal contrast for secret sharing schemes in visual cryptography. *Combinatorics, Probability and Computing*, 12, 285-299, 2003.

[BBS01] C. Blundo, A. De Bonis, and A. De Santis. Improved schemes for visual cryptography. *Designs, Codes, and Cryptography*, 24, 255-278, 2001.

[Poy03] Charles A. Poynton. *Digital Video and HDTV: Algorithms and Interfaces.* Morgan Kaufmann, 2003. Prieiga per internetą: http://books.google.lt/books?id=ra1lcAwgvq4C&pg=RA1-PA234&dq=wavelength+beams+additive

[DS11] R. Dastanian, H. S. Shahhoseini. Multi Secret Sharing Scheme for Encrypting Two Secret Images into Two Shares. *International Conference on Information and Electronics Engineering*, 2011.