VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
KOMPIUTERIJOS KATEDRA

Baigiamasis magistro darbas

Statistiniai stegoanalizės metodai

(Statistical Methods of Steganalysis)

Atliko: 2 kurso, 2 grupės studentė
Jekaterina Fedina (parašas)

Darbo vadovas:
doc. Vilius Stakėnas (parašas)

Vilnius
2012

Turinys

Anotacija

Pagrindinis šio darbo tikslas susipažinti su steganografijos mokslu bei statistiniais stegoanalizės metodais, kurių dėka slepiama ir atrandama informacija įvairiuose failuose. Šitame magistro darbe išnagrinėti, aprašyti bei įgyvendinti du steganografijos ir du stegoanalizės metodai. Visi metodai realizuoti JAVA programavimo kalba, daliai matematiniams skaičiavimams atlikti panaudota MAPLE programa. Darbo pabaigoje pateikta metodų analizė.

Summary

Statistical Methods of Steganalysis

The main goal of the thesis is to review the methods of steganography and steganalysis and to experiment with them. Steganography helps to embed hidden messages in such way that anyone except the intended recipient is unaware of the existence of the message but with the use of statistical steganalysis methods those hidden messages can be detected. This thesis consists of two steganography (LSB and LSBH) and two steganalysis methods (RS and PoV) description and implementation. All methods were implemented with JAVA code. Thesis is concluded with a comparison of these methods' quality.

Įvadas

Vienas is efektyviausių slaptos informacijos perdavimo būdų yra pacios informacijos egsistavimo paslėpimas. Informacijos slėpimo būdai naudoti labai seniai. Prasidėjus "skaitmeniniam" amžiui, atsirado steganografijos mokslas, padedantis įgyvendinti informacijos paslėpimą skaitmeniniuose objektuose. To pasekoje, atsirado ir stegoanalizė, kurios dėka galima aptikti paslėptą informaciją skaitmeniniuose objektuose.

Priešingai nei kriptografija, kuri užšifruoja informaciją, steganografija slepia pačios informacijos egzistavimą. Steganografija dažniausiai naudojama kartu su kriptografija papildant ją. Tokiu būdu padidėja tikimybė, kad slapto pranešimo nesužinos asmenys, kuriems ta informacija nėra skirta.

Prieš nagrinėjant bei realizuojant stegoanalizės metodus, būtina paruošti duomenis. Tam buvo įgyvendinti du steganografijos metodai. Kadangi steganografija nėra nauja sritis, šiuo metu galima surasti daugybę metodų. Šiame darbe nagrinėjami du steganografijos metodai – LSB (mažiausio reikšmingumo bito) ir LSBH (mažiausio reikšmingumo bito naudojant Hammingo kodą). Pirmas metodas yra gerai žinomas steganografijoje, antrojo metodo idėja yra originali.

Realizavus steganografijos metodus ir paruošus duomenis stegoanalizės metodams, pereiname prie statistinių stegoanalizės metodų, tokių kaip reguliari-vienetinė (RS) ir reikšmių poros (PoV). Abu metodai yra aprašyti ir įgyvendinti.

Pabaigoje pateikta metodų kokybės analizė. Pirmoji - steganografijos metodų analizė, pagal kurią galima spręsti, kuris metodas:

a) kokybiškiau slepia informaciją;
b) mažiau iškraipo originalų vaizdą;
c) yra tinkamas trumpiems/ilgiems stegotekstams.

Antroji - statistinių stegoanalizės metodų kokybės analizė, aprašanti kuris metodas:

a) turi didesnę tikimybę atrasti slapto stegoteksto egzistavimą;
b) geriau pritaikytas vienam arba kitam steganografijos metodui;
c) skirtas trumpiems/ilgiems stegotekstams.

1. Stegoanalizė ir jos istorija

Ne visada užtenka siunčiamą pranešimą tik užšifruoti, gana dažnai reikia paslėpti patį šifravimo faktą.

Steganografija – mokslas apie informacijos slėpimo būdus, stegoanalizė – mokslas apie tai, kaip aptikti paslėptą informaciją arba patį slėpimo faktą. Stegoanalitikas – asmuo, kuris bando perskaityti paslėptą tekstą atsiųstame pranešime. Tam atliekami trys žingsniai. Pirmiausiai, gavęs failą stegoanalitikas turi nustatyti, kuris steganografijos metodas buvo pritaikytas. Gautas failas apdorojamas ir sudaromas galimų steganografijos metodų sąrašas, kuriais galėtų būti užkoduotas pranešimas. Šio sąrašo dėka stegoanalitikas gali atskleisti slaptą pranešimą. Vėliau jis nagrinėja steganografijos metodų, kurios galėjo užkoduoti tą informaciją, ypatybes. Sėkmingai atpažinus ir išnagrinėjus steganografijos metodą, stegoanalitikas gali išgauti iš failo slaptą pranešimą.

Kaip ir duomenų kodavimas, taip ir steganografija laikoma vienu iš pagrindinių informacijos konfidencialumo saugojimo principų.

Manoma, kad steganografijos gimtinė yra Egiptas, nors steganografijos pranešimais būtų galima pavadinti ir senovės žmonių paveiksliukus ant sienų. Pats terminas „steganografija“ kilo iš graikų kalbos („stegonos“ - paslaptis ir „graphy“ – raštas) ir reiškia slaptaraštis, paslėptas raštas. XV amžiuje, steganografijos terminas pavartotas vokiečių autoriaus Johano Trihemijaus (Johannes Trithemius) išleistoje knygoje „Steganografija“ ([Gre09]). Steganografijai skirta daugelis slaptumo įrankių tokių, kaip permatomas rašalas, mažos nuotraukos, paslaptingi kanalai, ženklų pasiskirstymas ir t.t.

Steganografija yra mažiau svarbi duomenų saugojimo atžvilgiu, nes ji ne pakeičia, o papildo kriptografiją. Pranešimo slėpimas steganografijos metodais sumažina tikimybę jį aptikti, tačiau jeigu pranešimas dar bus ir užšifruotas kriptografijos metodais, tai saugumo lygis smarkiai padidės.

Kaip ir dauguma saugumo įrankių, steganografija gali būti naudojama skirtingiems tikslams. Tai gali būti:

- apsauginiai ženklai ant valiutų;
- skaitmeniniai parašai, tokie kaip pirštų antspaudai, kurie patvirtina žmogaus tapatumą, autorines teises. Kalbant apie pirštų antspaudus, tai vienas iš geriausių steganografijos principų, kadangi atspaudas lieka ant paliestų daiktų ir nepastebimas žmogaus akimi;

- podėlio (angl. Cache) reikšmės keitimas - imamas įvesto kintamojo ilgis ir keičiamas į statinį eilutės ilgį tam, kad būtų patvirtinta jog įvestas kintamasis nebuvo pakeistas vykdymo metu;
- parašas, patvirtantis autorines teises, ant internete patalpinto paveiksliuko;
- slaptos informacijos saugojimas tam, kad būtų apsisaugota nuo vagysčių ir nesankcionuotos peržiūros.

Tačiau steganografija yra naudojama ir neteisėtiems veiksmams. Pavyzdžiui, bandant išsiųsti pavogtą informaciją, ji įrašoma kitame faile arba failuose ir elektroniniu paštu siunčiama kaip paprastas failas ar paveiksliukas. Be to steganografiją galima naudoti, slepiant informaciją kietajame diske arba norint slaptai bendrauti su tam tikrais žmonėmis.

Dažnai steganografija yra naudojama kartu su kriptografija. Kriptografija pagalba stengiamasi apsaugoti pranešimą nuo įsilaužėlių ir pristatyti jį tik tam asmeniui, kuriam jis buvo skirtas. Tuo tarpu steganografija slepia ne tik siunčiamą pranešimą, bet ir patį pranešimo slėpimo faktą. Dėl šių priežasčių kartais yra efektyviau tas dvi metodikas naudoti kartu. Steganografija sukuria saugų persiuntimo kanalą, o kriptografija – duomenų apsaugą. Tokiu būdu pranešimas nukeliaus saugiu kanalu.

Steganografijos metodika negali būti tiesiogiai priskirta kriptografijai. Jų bendras bruožas yra tai, kad abi metodikos stengiasi kiek įmanoma daugiau išlaikyti originalios informacijos, tai yra, kuo mažiau ją iškraipyti.

Didėjant žmonių priklausomybei nuo kompiuterių, steganografija tampa vis populiaresnė slepiant skaitmeninius duomenis. Dauguma failų, elektroninių dokumentų, garso ir vaizdo įrašų turi kažkiek neišnaudotų bitų. Šiuolaikinė steganografijos technologija išnaudoja šių tuščių informacijos plotų galimybes, įrašant tam tikrą informaciją į tuščias vietas arba pakeičiant dalį originalo informacijos žmogui negirdima, nematoma informacija. Steganografijos būdu, dažniausiai informacija slepiama paveikslėliuose ir garso failuose, iš pradžių užšifruojant, o po to įdiegiant į slepiantį vaizdą. Pranešimas gali būti paprastas tekstas arba kitas vaizdas. Sujungus slepiantį vaizdą ir slepiamą pranešimą, gaunamas stegovaizdas. Stegorakto pagalba, kuris yra įvedamas specialia steganografijos programine įranga, pranešimas yra paslepiamas arba atidengiamas. Tik tas asmuo, kuris žino kaip yra įdiegtas pranešimas gali iššifruoti ir perskaityti jį. Steganografija skirstoma į tris kryptis ([Wiki11a]):

- klasikinė steganografija;
- kompiuterinė steganografija;
- skaitmeninė steganografija.

Toliau trumpai apžvelgsime kiekvieną kryptį.

1.1. Klasikinė steganografija

Laikoma, kad senovės šumerai vieni iš pirmųjų pradėjo naudoti steganografiją, nes buvo rasta daug dantiraščio lentelių iš molio ant kurių vienas užrašas buvo slepiamas po molio sluoksniu, o ant antro sluoksnio buvo užrašytas kitas užrašas. Tačiau istorikai tikrina, kad tai nebuvo slėpimas, o tik praktiškas panaudojimas.

Du informacijos slėpimo metodus mini graikų istorikas Herodotas:

1. Ant skustos vergo galvos buvo rašomas pranešimas, kurį reikėjo perduoti asmeniui iš kito krašto, ir buvo laukiama kol plaukai ataugs. Tada vergas buvo siunčiamas pas adresatą, kuris vėl nuskusdavo jam galvą ir perskaitydavo pranešimą.
2. Pranešimas buvo užrašomas ant medinės lentelės, o tada lentelė buvo tepama vašku. Vėliau tas vaškas buvo nuskutamas ir pranešimas perskaitomas ([Gre09]).

Pagrindinė, klasikinio stiliaus, steganografija naudoja permatomus rašalus, kurie matomais tampa tik prie tam tikrų sąlygų: šilumos, apšvietimo, cheminių medžiagų ir t.t.

II-ojo Pasaulinio karo metu aktyviai buvo naudojamas mikrotaškiukas, kurio diametras ne didesnis nei vienas milimetras – mikroskopinės nuotraukos buvo įklijuojamos į tekstą laiške arba telegramose.

Šaltiniuose minimi ir kiti slėpimo metodai:

- užrašas ant kortų kaladės šono;
- užrašas virto kiaušinio viduje;
- darkyti žodžiai, kurie turi kita reikšmę;
- trafaretai, kurie uždedant ant teksto, palieka matomas tik tas raides, iš kurių susideda pranešimas;
- mazgai ant siūlų ir t.t ([Wiki11a]).

Šiuo metu, informacijos slėpimas tekstiniuose, vaizdiniuose ar garso failuose, realizuojamas specialios programinės įrangos pagalba.

1.2. Kompiuterinė steganografija

Kompiuterinė steganografija – klasikinės steganografijos šaka, realizuota kompiuterinėje aplinkoje. Šiuolaikinis gyvenimas neatsiejamas nuo kompiuterinių naujovių, o su jomis atsiranda

slaptumo bei saugumo reikalaujantys subjektai, todėl buvo sukurti kompiuterinės steganografijos metodai. Kaip pavyzdys gali būti StegFS (Steganographic File System) programa, kuri skirta informacijos slėpimui nenaudojamuose failų registruose, simbolių pakeitimui failų pavadinimuose, tekstinėse steganografijose ir t.t ([Wiki11a]). Slepiamais duomenimis gali būti bet kokio formato informacija: paveiksliukas, vaizdo ar garso failas, ar paprastas tekstas.

Žiūrint į dabartinę situaciją ir globalų augimą kompiuterijos sferoje, steganografija tampa vis populiaresnė.

1.3. Skaitmeninė steganografija

Skaitmeninė steganografija – taip pat klasikinės steganografijos šaka, sukurta slėpti informaciją įterpiant ją į skaitmeninius objektus. Tokiais objektais gali būti paveiksliukai, nuotraukos, vaizdo, garso, 3D-objektų tekstūra. Nukrypimas egzistuoja, tačiau nedidelis, žmogaus akimi sunkiai pastebimas.

Skaitmeninė steganografija, priešingai nei kompiuterinė, neįterpia papildomų duomenų į interneto protokolo antraštes, skirtingų failų formatus ar tekstinius pranešimus. Didžiąja dalimi, skaitmeninė steganografija yra skirta konfidencialios informacijos įterpimui ir skaitmeninių ženklų įterpimui į statinius objektus, pavyzdžiui, į failus, kurie nenaudoja duomenų suspaudimo (*.BMP). Tačiau dabar atsiranda vis daugiau algoritmų skirtų grafiniams failams su duomenų suspaudimu, tokiems kaip JPEG ([Wiki11a]).

Skaitmeninių ženklų metodas buvo sukurtas kaip legalus skaitmeninės steganografijos metodas, skirtas autorinių teisių sistemai. Pavyzdžiui, Adobe Photoshop programa leidžia į paveiksliuką įdėti informaciją apie autorių. Tačiau šitas ženklas nėra gerai apsaugotas ir jį galima panaikinti kitomis programomis.

Skaitmeninės steganografijos algoritmus galima suskirstyti į kelias grupės:

- darbo su skaitmeniniu signalu, tokiu kaip LSB (Least Significant Bit), metodas;
- slaptos informacijos įterpimas į originalų failą (dažniausiai naudojamas skaitmeninių ženklų metodas);
- naudojimasis specialiais failų formatais, t.y. meta-duomenų įrašymas į nepanaudotus rezervuotus failo laukus ([Wiki11a]).

Įterpimui naudojami skirtingi steganografijos algoritmai.

2. Metodai, naudojantys mažiausio reikšmingumo bitus

Pagrindinė darbo tema yra stegoanalizė, tačiau prieš nagrinėjant bei realizuojant stegoanalizės statistinius metodus, reikėtų paruošti duomenis atpažinimui. Tam reikėtų išnagrinėti steganografijos metodus ir jų dėka paslėpti informaciją nagrinėjamame objekte.

Iš daugybės steganografijos metodų buvo pasirinkti du:

- LSB metodas (*Least Significant Bit* iš anglų kalbos - mažiausio reikšmingumo bitas) - vienas iš plačiausiai naudojamų;
- Hammingo kodo struktūra pagrįstas mažiausio reikšmingumo bito metodas (LSBH).

Viena iš svarbiausių steganografijos užduočių yra teisingai parinkti tinkamą failą. Tam buvo sukurti pagrindiniai kriterijai parenkant failą. Vieni iš svarbiausių aspektų yra:

- triukšmas;
- staigūs spalvos perėjimai;
- erdvių monotoniškumo nebuvimas;
- didelis spalvų ryškumo šuolių kiekis.

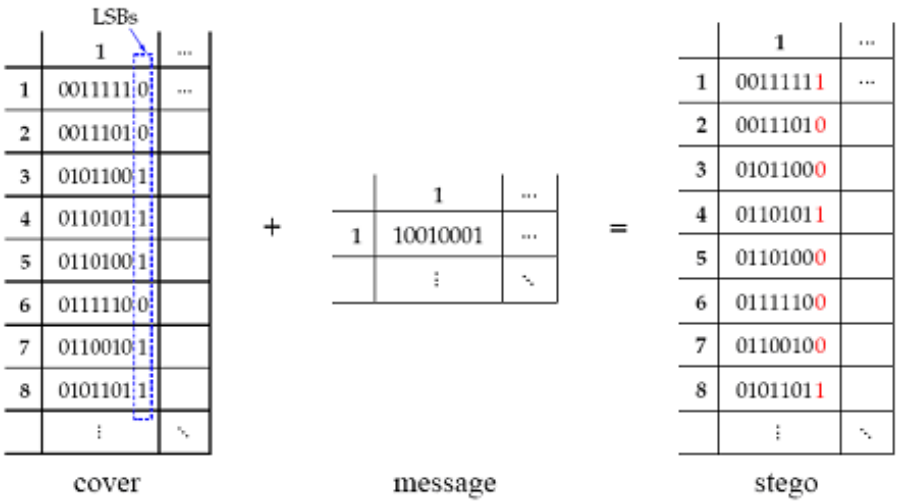
Nuo kokybiškai parinkto failo priklauso slapto pranešimo ilgis bei pranešimo paslėpimo atskleidimo tikimybė. Dėl to yra labai svarbu panaudoti tinkamą metodą paveiksliuko parinkimui tam, kad slaptą informaciją faile būtų sunku aptikti stegoanalizės metodais.

Metodai yra aprašyti bei įgyvendinti, naudojant JAVA programavimo kalbą. Nagrinėjamu objektu buvo pasirinktas failo formatas, kuris nenaudoja duomenų suspaudimo, (*.BMP). Slepiamas pranešimas (stegotekstas) yra skirtingo tipo duomenys: kiti paveiksliukai, tekstai bei atsitiktinė bitų seka.

2.1. Mažiausio reikšmingumo bito metodas (LSB)

LSB (*Least Significant Bit* iš anglų kalbos - mažiausio reikšmingumo bitas) – vienas iš metodų skirtas slėpti informaciją statiniuose failų formatuose. Jo pagrindu realizuota daugelis programų skirtų slėpti pranešimus paveiksliukuose, kurie skirti naudojimui žiniatinklyje.

Šitas metodas dirba su mažiausio reikšmingumo bitais paveikslėlyje. Mažiausio reikšmingumo bitai pakeičiami slapto pranešimo bitais taip, kad rezultate nesimatytų pranešimo buvimas. Patartina keisti tik tuos bitus, kurie neįeina į paveiksliuko vaizdo atkūrimą (2.1. pav. [Rgi11]).



2.1 pav. [Rgi11]

Tokiu būdu žmogaus akims paveiksliukas liks nepasikeitęs, nebus juntamas skirtumas tarp originalaus ir pakeisto paveiksliuko, tačiau statistiniai paveiksliuko parametrai bus pakeisti. Dažniausiai tokius paveiksliukus papildomai suspaudžia arba archyvuoja tam, kad jo apimtis sumažėtų, o parametrai taptų artimesni atsitiktiniams parametrams.

Stegoteksto įterpimo atveju, t.y. keičiant LSB į atsitiktinį eiliškumą, pikselių kiekis porose tampa lygus, o histograma - laiptuota. Stegoanalizės metodai, lygindami tokias histogramas gali aptikti slepiamus bitus.

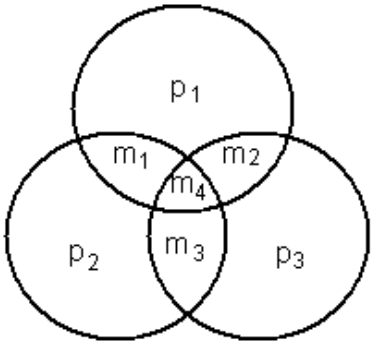
LSB metodas nėra apsaugotas nuo įvairių stegoanalizės metodų, todėl patartina naudoti tik tiems paveiksliukams, kurie turi triukšmą.

2.2. Mažiausio reikšmingumo bito metodas naudojant Hammingo kodą (LSBH)

Amerikiečių matematikas Ričardas Hammingas (*Richard W. Hamming*) dirbo *Bells Labs* (*Bell Labaratories*, ankstesni pavadinimai - *AT&T Bell Laboratories, Bell Telephone Laboratories*) kompanijoje su *Bell Model V* skaičiavimo mašina. Jis dažnai budavo susierzinęs ir praleisdavo savaitgalius darbe, nes jam pernelyg dažnai tekdavo perkrauti savo programą dėl perforacinės kortos nepatikimumo. Keletą metų jis praleido kurdamas efektyvesnį klaidų taisymo algoritmą. 1950 metais jis paskelbė savo sukurtą metodą, kuris dabar yra garsus kaip Hammingo kodas (*Hamming Code*) ([Sta06], [Wiki11b]).

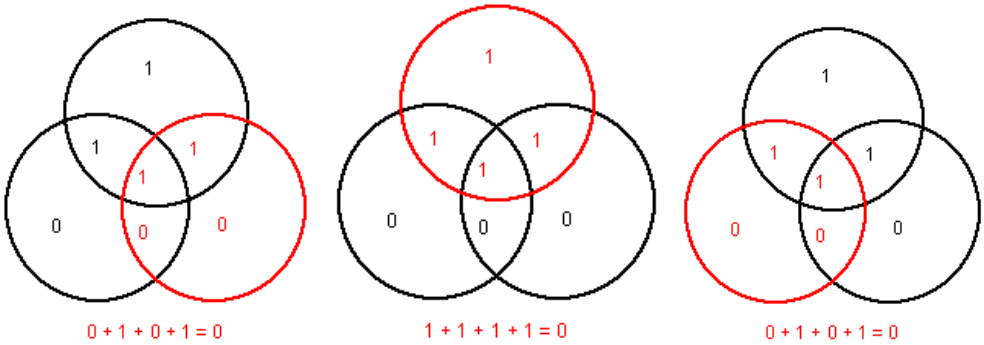
Hammingo pasiūlytas algoritmas – tai specifinis algoritmo tipas skirtas atrasti bei pataisyti klaidingus bitus, kitaip tariant – klaidas taisantis kodas [Dam03]. Jo principas: užkoduoti 4 bitus

(m_1, m_2, m_3, m_4) į 7 segmentus pridedant tris kontrolinius (p_1, p_2, p_3) bitus (2.2.1. pav. [Wiki11a]).



2.2.1. pav. [Sta06]

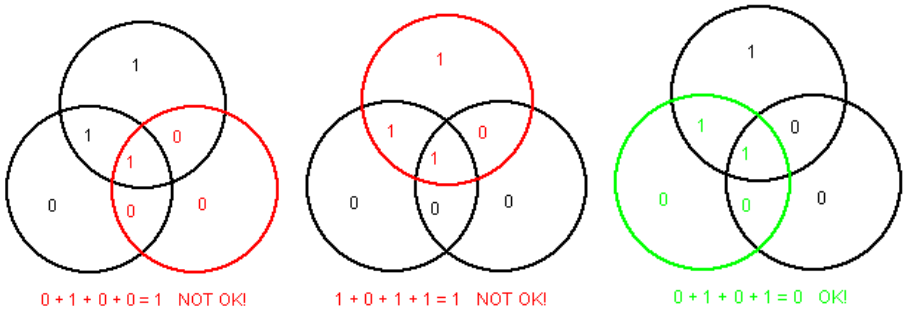
„Kiekvienas kontrolinis bitas parinktas taip, kad bendras vienetų skaičius jo apskritime yra lyginis.“ [Dam03].



2.2.2. pav. [PA03]

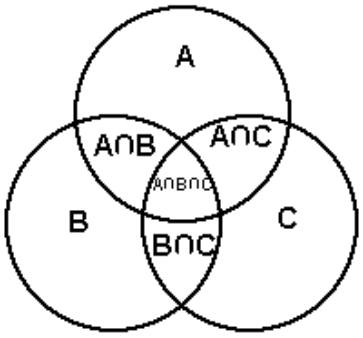
Žiūrint į pavyzdį viršuje (2.2.2. pav. [PA03]), galima pamatyti teisingus bitus, kai buvo koduojami bitai M = (m_1, m_2, m_3, m_4) = (1, 1, 0, 1) ir kontroliniai bitai K = (p_1, p_2, p_3) = (1, 0, 0). Jeigu visų apskritimų sumos lygios 0, reiškia kad bitas yra teisingas.

Tačiau jeigu koduojami bitai M = (m_1, m_2, m_3, m_4) = (1, 0, 0, 1) ir kontroliniai bitai K = (p_1, p_2, p_3) = (1, 0, 0), tai dviejų apskritimų bitų sumos bus nelyginės (2.2.3. pav. [PA03]).



2.2.3. pav. [PA03]

Kadangi dviejų apskritimų bitų sumos yra nelyginės, tai galima daryti išvada, kad m_2 bitas yra klaidingas. Tai galima apskaičiuot naudojant sankirtą (2.2.4. pav.).



2.2.4. pav.

Hammingo (7,4) kodas gali atrasti ir pataisyti klaidingą bitą. Pridedant kontrolinius bitus gali būti aptikti (tačiau ne taisomi) ir du neteisingi bitai. Taigi visa kodo ypatybė yra ta, kad klaidingai pasikeitus kuriam nors žodžio bitui, galima atrasti klaidingo bito numerį tinkamai įdedant kontrolinius bitus (netgi tuo atveju, kai klaida yra kontroliniame bite). Hammingo kodas leidžia ne tik aptikti vieną klaidą, bet ir ištaisyti ją.

Panaudojus šitą principą steganografijoje, galima užslėpti informaciją mažiausio reikšmingumo bite. Taigi tam buvo sukurtas mažiausio reikšmingumo bito metodas naudojant Hammingo kodą (LSBH - Least Significant Bit Hamming) ([Sta06]).

Metodo realizavimui paimami bitai, kurie turi būti paslėpti $s = s_1, s_2, s_3, ...$ Tada imamas pirmasis konteinerio baitas, naudojant tik pirmuosius septynis jo bitus $x_1, x_2, x_3, x_4, x_5, x_6, x_7$ (čia x_1 mažiausio reikšmingumo bitas). Septynių bitų žodis yra $x = x_1x_2x_3x_4x_5x_6x_7$ interpretuojamas kaip Hammingo kodo žodis (gali būti iškraipytas). Patikrinama, ar jis neiškraipytas išskaidant jį į 7 sektorius ($m_1, m_2, m_3, m_4, p_2, p_1, p_3$) (2.2.1. pav. [Wiki11a]) ir patikrinant baito teisingumą, naudojant tris formules:

- $p_1 \cap m_1 \cap m_2 \cap m_3 = 0$,
- $p_2 \cap m_1 \cap m_2 \cap m_4 = 0$,
- $p_3 \cap m_1 \cap m_3 \cap m_4 = 0$.

Jeigu kažkuris bitas yra neteisingas, tai reikia jį surasti, o tada naudoti taisykles, tokias kaip (žiūrėkite „2.2.1. pav.“ schemą):

1. Jeigu baitas yra neiškraipytas, tikrinami x_6 ir x_7 (p_1 ir p_3) bitai su dviem steganografijos bitais (s_1s_2):

a) jeigu poros sutampa, tada laikoma, kad du konteinerio baito bitai slepia du stegoteksto bitus s_1s_2 ir imamas sekantis konteinerio baitas.
b) jeigu nesutampa, x_1 (m_4) pakeičiamas priešingu, taip sugadindamas baitą. Imamas sekantis konteinerio baitas.

2. Jeigu baitas iškraipytas ir sugadintas yra vienas iš x_2, x_3, x_4 (m_1, m_2, m_3) bitų, tada tikrinama:
   a) jei stegoteksto bitas lygus 0, tada laikoma, kad pakeistas baitas slepia 0 ir x_1 (m_4) keičiamas priešingu bitu. Imamas sekantis konteinerio baitas.
   b) jei stegoteksto bitas lygus 1, tada niekas nėra keičiama. Imamas sekantis konteinerio baitas.
3. Jeigu iškraipytas ir sugadintas baitas yra vienas iš x_5, x_6, x_7 (p_3, p_1, p_3) bitų, tikrinama:
   a) jei stegoteksto bitas lygus 0, tada niekas nėra keičiama. Imamas sekantis konteinerio baitas.
   b) jei stegoteksto bitas lygus 1, tada laikoma, kad pakeistas baitas slepia 1 ir tada x_1 (m_4) keičiamas priešingu bitu. Imamas sekantis konteinerio baitas.
4. Jeigu iškraipytas ir sugadintas baitas yra x_1 (m_4) bitas:
   a) niekas nėra keičiama, t.y. laikoma, kad baitas neslepia jokios informacijos. Imamas sekantis konteinerio baitas.

Analogiškai slepiami ir kiti sekos s bitai. Tokiu būdu gaunama, kad kai kurie konteinerio baitai bus iškraipyti, tačiau nustatyti, kuris iš bitų yra klaidingas gali tik tas asmuo, kuris žinos šitas sąlygas.

Atkuriant paslėptą informaciją, skaitomas kiekvienas baitas, tikrinama jo iškraipytojo baito pozicija ir naudojant taisykles bus nustatomas stegoteksto bitas:

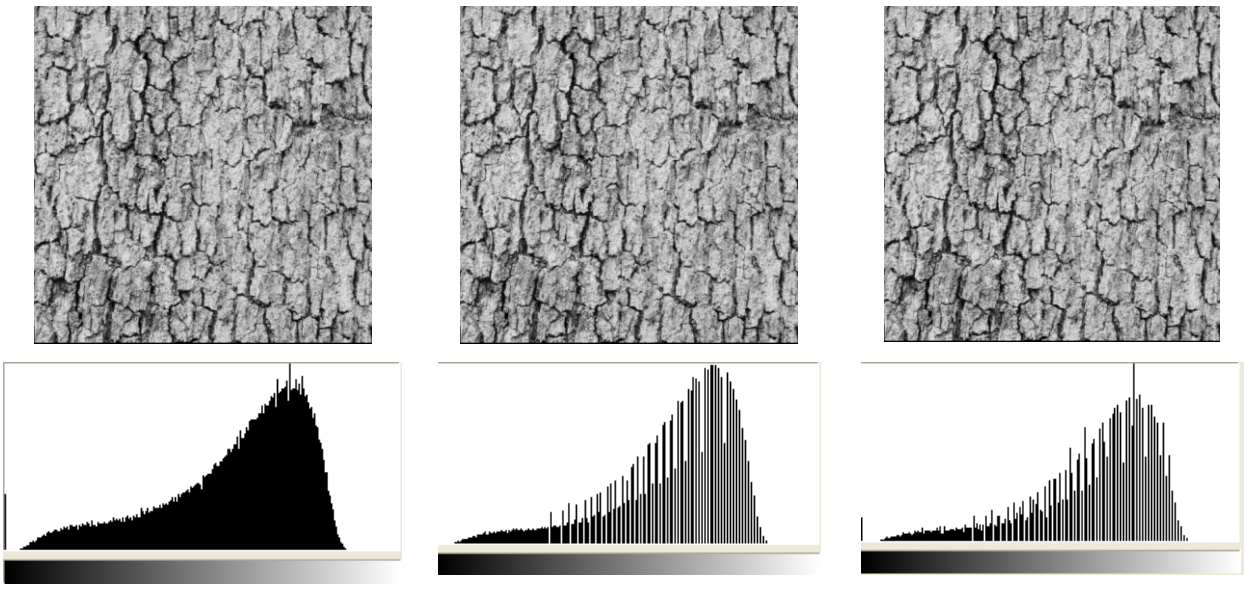
1. Jeigu konteinerio baitas yra neiškraipytas:
   a) reiškia du paskutiniai konteinerio baitai slepia du stegoteksto bitus.
2. Jeigu konteinerio baitas yra iškraipytas ir sugadintas yra vienas iš x_2, x_3, x_4 (m_1, m_2, m_3) bitų:
   a) reiškia stegoteksto bitas lygus 1.
3. Jeigu konteinerio baitas yra iškraipytas ir sugadintas yra vienas iš x_5, x_6, x_7 (p_3, p_1, p_3) bitų:
   a) reiškia stegoteksto bitas lygus 0.
4. Jeigu konteinerio baitas yra iškraipytas ir sugadintas yra x_1 (m_4):
   a) reiškia baitas neslepia stegoteksto bitų.

Taip pereinamas visas paveiksliukas ir perskaitomas stegotekstas. Tam kad nustatyti stegoteksto ilgį, šifruojant galima į pradžią įdėti stegoteksto ilgį arba stegoteksto galę uždėti pabaigos simbolį.

3. Steganografijos metodų realizacija bei rezultatai

Sukurtoje programoje buvo panaudoti du aukščiau išnagrinėti metodai. Konteineriu buvo pasirinktas nespalvotas paveiksliukas BMP (*BitMap Picture* – taškinės grafikos skaitmeninis paveiksliukas) formato. Steganografijos subjektas – tekstas arba nespalvoti paveiksliukai BMP formatu. Programavimo kalba – JAVA.

Rezultate, panaudojus 80 000 bitų tekstą paveiksliukui 512 x 512, gaunama (3.1. pav.):

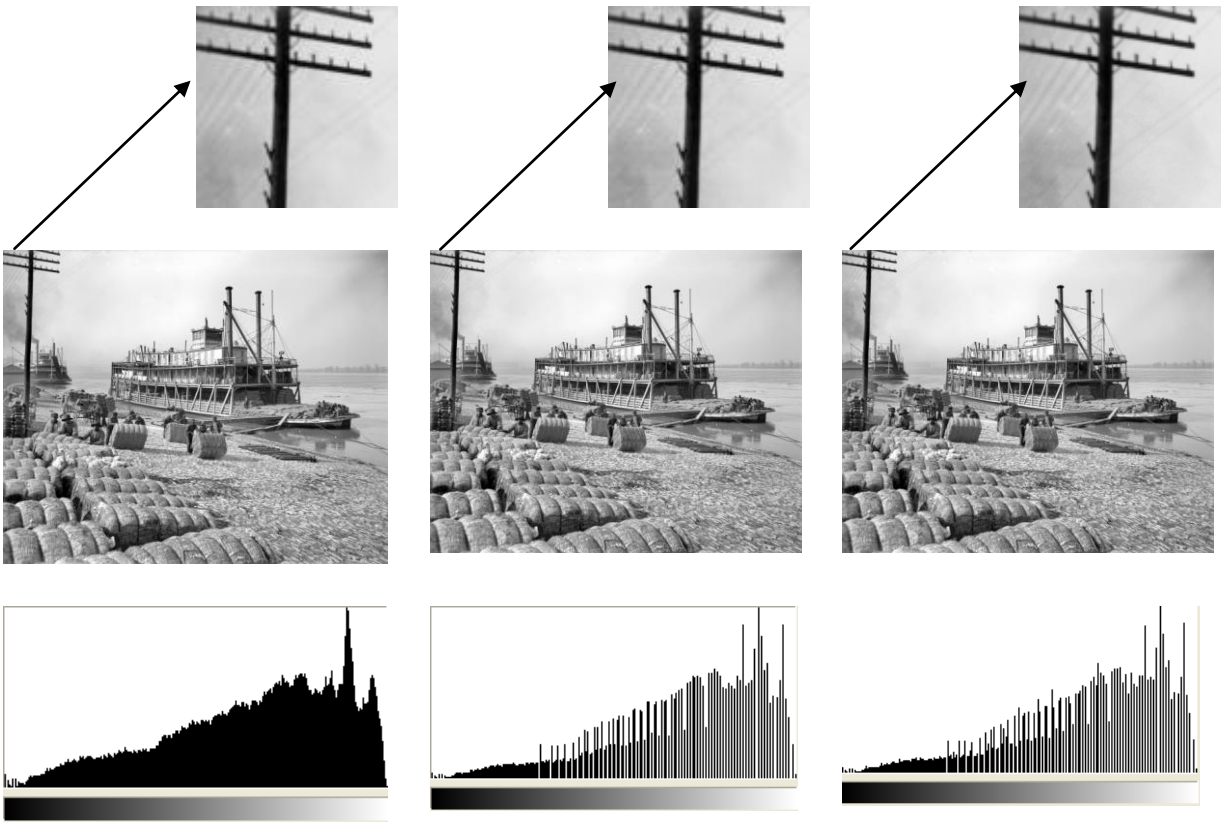


Originalas LSB LSBH

3.1. pav.

Jokių skirtumų paveiksliukuose nesimato, nors LSB metodu buvo pakeista 50,15 % baitų iš 80 000 peržiūrėtų (tai ~28 % viso paveiksliuko baitų), o Hammingo kodo metodu – 47,40 % baitų iš 100 295 peržiūrėtų (tai ~38 % viso paveiksliuko baitų). Tačiau jeigu pažiūrėsime į paveiksliukų histogramas, galima pastebėti pasikeitimus (3.1. pav.).

Kito bandymo metu, paimtas paveiksliukas su didesniu elementų kiekiu (591 x 480), tai yra paveiksliukas, kuris turi daugiau atvaizdo detalių. Užkoduojant jame 80 000 bitų stegotekstą, gaunamas rezultatas pavaizduotas 3.2. pav.:



Originalas LSB LSBH

3.2. pav.

Vizualiai, paveiksliukuose nesimato jokių skirtumų žmogaus akimi, nors LSB metodu buvo pakeista net 49.84 % baitų iš 80 000 peržiūrėtų (tai ~28 % viso paveikslio baitų), o LSBH metodu – 47.92 % baitų iš 97 041 peržiūrėtų (tai ~34 % viso paveikslio baitų). Tačiau pažiūrėjus į histogramas, matomi aiškūs pakeitimai paveiksliukuose.

Rezultatų suvestinė, slepiant 100 295 bitų stegotekstą turint skirtingą kiekį paveiksliukų, yra tokia (3.3. lentelė):

Kiekis		% pakeistų bitų	
Paveiksliukų	Stegotekstų	**LSB**	**LSBH**
1	500	50,32	45,01
500	1	50,16	45,59

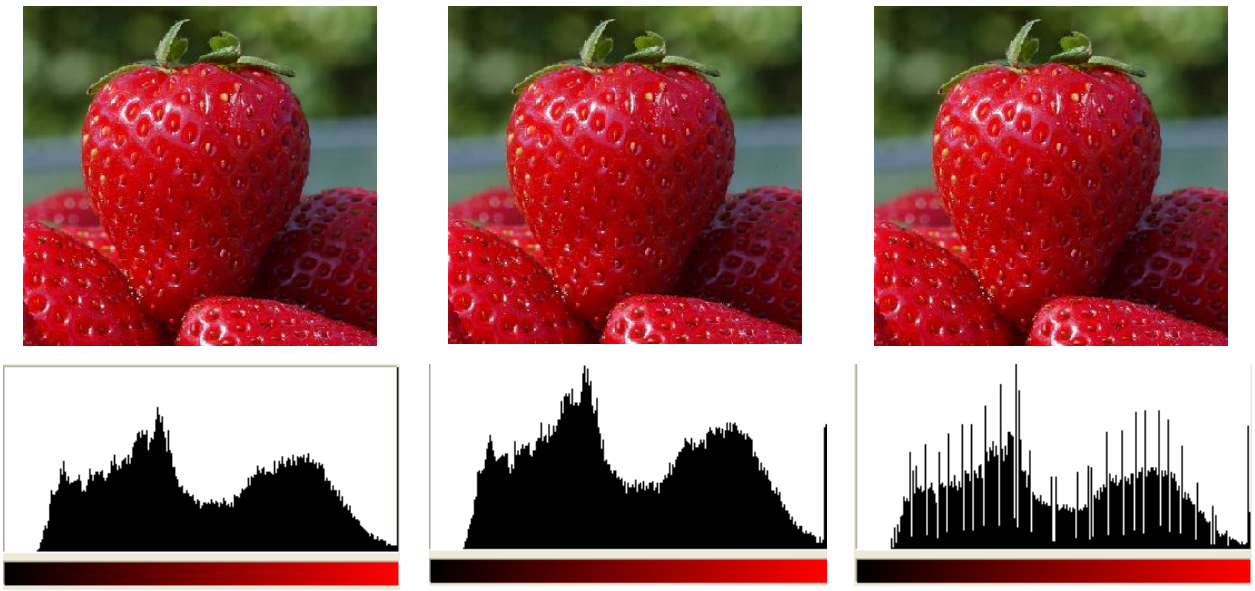
3.3. lentelė

Jeigu į vieną paveiksliuką bus įdėta 500 skirtingo ilgio ir tipo stegotekstų (tekstas, paveiksliukas, atsitiktinė bitų seka), bet stegoteksto ilgio *bitais* skaičius bus per pusę mažesnis už paveikslio dydžio *baitais* skaičių (nes buvo pastebėta, kad LSBH metodas daug baitų

praleidžia nekeisdamas, dėl to kad sutinka iškraipytą baitą), tai gaunamas rezultatas, kad LSB metodas pakeičia apie 50,32 % baitų, o LSBH – iki 5 % mažiau. Kitame variante buvo paimta daug skirtingo dydžio paveiksliukų ir vienas stegotekstas. Gaunamas rezultatas - LSBH metodas taip pat iškraipo mažiau baitų negu LSB metodas.

Taip pat buvo atlikta analizė su spalvotais BMP formato paveiksliukais. Buvo paimtas spalvotas 600 x 800 pikselių paveiksliukas ir įdėtas 480 000 bitų dydžio (viso paveiksliuko dydžio baitais skaičius) stegotekstas LSB ir LSBH metodais. Buvo keičiami konteinerio raudonos spalvos mažiausio reikšmingumo bitai.

Rezultatas gavosi panašus į rezultatą gautą apdorojant nespalvotus paveiksliukus: vaizdas žmogaus akimis nepasikeitė, bet histogramose pastebimas skirtumus (3.4. pav.).



Originalas LSB LSBH

3.4. pav.

Paveiksliuke buvo pakeista 50,18 % baitų LSB metodu ir 46,84 % baitų LSBH metodu. Pažiūrėjus į histogramas, matome, kad LSB metodu pačios histogramos aukštis tolygiai pakylo, o štai LSBH metodu matomas netolygus histogramos išsiskirstymas.

Streganografijos metodų analizei buvo parinkta virš 300 spalvotų 600 x 800 pikselių dydžio paveiksliukų, į kuriuos buvo bandoma įterpti skirtingo dydžio stegotekstus. Jeigu paveiksliuką sudaro 10 000 baitų, tai stegoteksto ilgis bus 5 000 bitų (50 %), 2 500 bitų (25 %).

Įterpti virš 50 % stegoteksto ilgio negalima, kadangi LSBH metodas slepia informacija ne kiekviename baite, todėl šio metodo eigoje yra pereinama daugiau baitų, nei stegoteksto ilgis. Jeigu paimsime 600 x 800 = 480 000 baitų paveiksliuką ir į ji bandysime įterpti 50 % ilgio

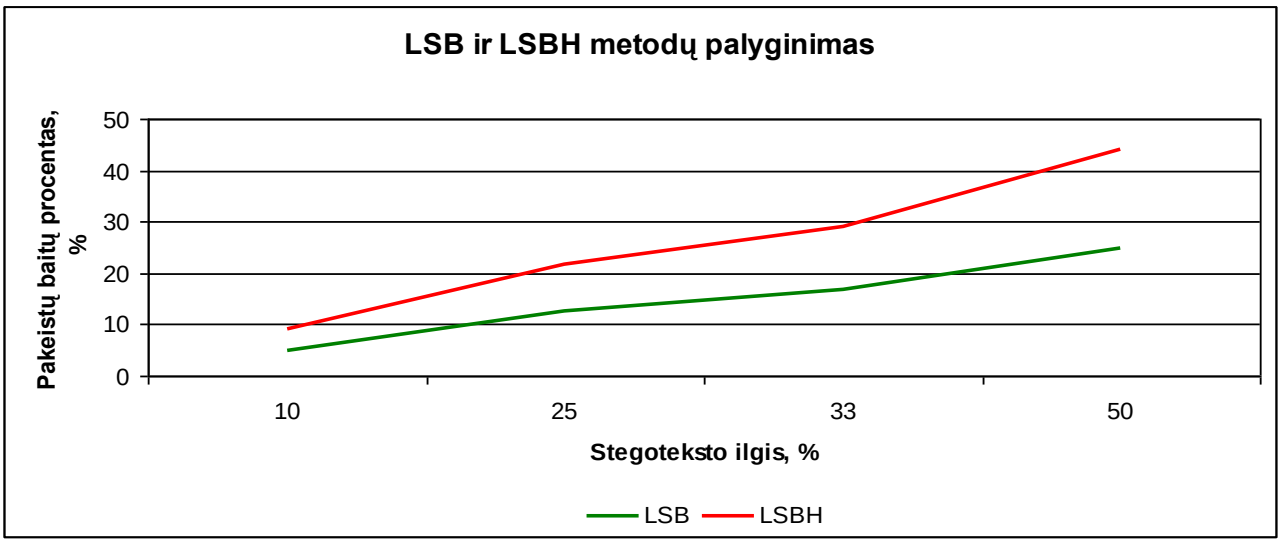
stegotekstą (240 000 bitų), tai LSB metodu bus peržiūrėta 50 % baitų, o LSBH metodu – 80 %, o tai yra net 30 % daugiau, nei LSB metodu.

Stegotekstu buvo parinkta atsitiktinė bitų seka, kadangi taip yra lengviau nustatyti stegoteksto ilgį, be to, paveiksliuke būna pakeičiama daugiau bitų, o nuo to gali pasikeisti paveikslio vaizdas. Tuo siekiama geriau įvertinti ir palyginti steganografijos metodus.

Buvo gautas rezultatas, kad abu metodai gerai slepia stegotekstą, t.y. iškraipo nedaug bitų. Taip pat galima pastebėti, kad geriau stegotekstas yra slepiamas LSB metodu, nes sugadina iki 10 % mažiau baitų, nei LSBH metodas (3.5. lentelė, 3.6. diagrama). Tačiau LSBH metodas perskaito daugiau baitų, nei LSB metodas, o tai sumažina galimybę įterpti ilgesnį stegotekstą.

Stegoteksto ilgis, %	Pakeistų baitų, %	
	LSB	**LSBH**
10	5,01	8,98
25	12,47	21,78
33	16,64	28,99
50	25,00	44,04

3.5. lentelė



3.6. diagrama

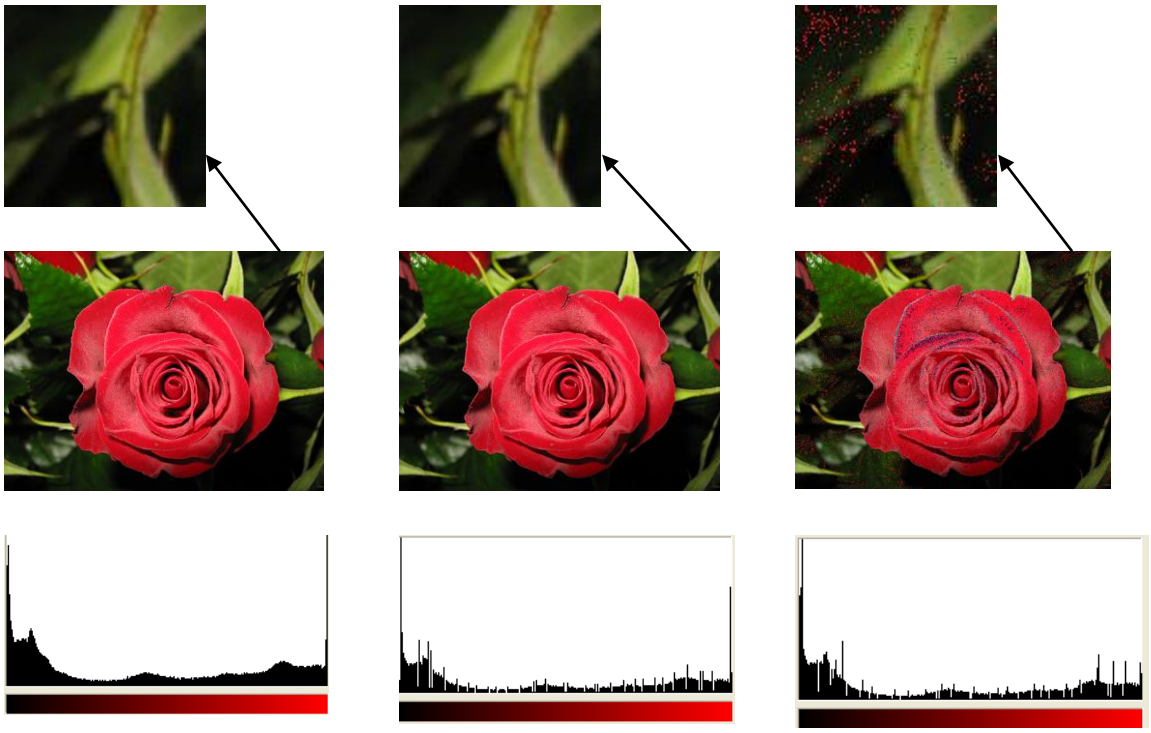
Taigi, galima teigti, kad abu metodai LSB ir LSBH gerai slepia informaciją nespalvuotuose bei spalvotuose paveiksliukuose. Tik specialus steganografijos metodai ir žmonių budrumas gali pastebėti iškraipytą paveiksliuką. Be to galima pabrėžti, kad iš šitų dviejų metodų, nespalvotiems paveiksliukams geresnis yra LSBH, kadangi jo algoritmas sugadina mažiau baitų, tačiau žiūrint į

histogramas – spalvotiems geresnis yra LSB metodas. Jo dėka yra mažesnė tikimybė, kad žmogus pastebės netikslumus konteineryje.

Pastebėta, kad LSBH metodas tikrina daugiau baitų, nei LSB metodas, todėl buvo bandoma patobulinti šį aspketą. Analizei buvo bandoma patobulinti LSBH metodą. Idėjos tikslas - išnaudoti neiškraipytą baitą. Tam patobulinamas 14psl., 1.b punktas: Jeigu baitas yra neiškraipytas ir pirmi du stegoteksto bitai nesutampa su paveiksliuko paskutiniais dviems bitais, tai:

a) jeigu slepiamo bito reikšmė yra lygi 1, tai pakeisti x_2 bitą priešingu;
b) jeigu slepiamo bito reikšmė yra lygi 0, tai pakeisti x_5 bitą bitą priešingu.

Tokiu būdu, atkuriant paslėptą informaciją, daugiau baitų bus iškraipyta, tačiau apdorota mažiau (tiek pat kiek ir LSB metodu) ir kiekvienas iš jų neš informaciją. Patestavus šitą patobulinimą buvo padaryta išvada, kad LSBH metodas skanuoja žymiai mažiau baitų ir sugadintų baitų procentas sumažėja. Tačiau, metodo kokybė pablogėja. Tokiu būdu paveiksliukų spalvos išsikraipomos ir pasikeitimas matomas net žmogaus akimi, kadangi keičiami yra ne tik mažiausio reikšmingumo bitai. Tolimesnėse analizėse šitas patobulinimas nebuvo naudojamas (3.8. pav.).



Originalas | LSBH (pakeista 65,79 %) | LSBH + patobulinimas (pakeista 61,25 %)

3.8. pav.

3.1. Mažiausio reikšmingumo bito metodo realizavimo rezultatai

Sukurtos programos aprašymas: programai paduodama skirtingų dydžių nespalvotų bei spalvotų BMP formato paveiksliukų biblioteka ir stegotekstai, kurie turi būti paslėpti paveiksliukuose. Steganografijos tipas gali būti tekstas, nespalvotas BMP formato paveiksliukas arba atsitiktinė bitų seka.

Pateiktas stegotekstas skaidomas po bitą. Tam, kad patikrinti ar stegoteksto dydis yra tinkamas, perskaitomas paveiksliukas ir apskaičiuojamas jo dydis. Jeigu steganografijos dydis tinka šiam paveiksliukui, tada vykdomas bitų patikrinimas (paskutinis kiekvieno paveiksliuko baito bitas su kiekvienu stegoteksto bitu). Jeigu atrandami nesutapimai – paveiksliuko bitas keičiamas stegoteksto bitu. Tokiu būdu stegotekstas yra slepiamas mažiausio reikšmingumo bite.

Taip yra kartojama su visais paveiksliukais ir stegotekstais.

Rezultatas (t.y., paslėptas stegotekstas), kaip ir buvo pateikta teorijoje, yra nepastebimas žmogaus akimi. Tinkamai parinktame paveiksliuke nesimato pakeitimų, tačiau stegotekstas jame yra paslėptas. Tai ir buvo šio darbo tikslas. Slepiant stegotekstą LSB metodu problemų neiškilo.

3.2. Mažiausio reikšmingumo bito metodo naudojant Hammingo kodą realizavimo rezultatai

Sukurtos programos aprašymas: programai paduodama skirtingų dydžių nespalvotų arba spalvotų BMP formato paveiksliukų biblioteka ir stegotekstai, kurie turi būti paslėpti paveiksliukuose. Slapto pranešimo tipas gali būti tekstas, nespalvotas BMP formato paveiksliukas arba atsitiktinė bitų seka. Stegoteksto dydis turi būti mažesnis nei pusė konteinerio dydžio, kadangi LSBH metodas slepia stegotekstą ne kiekviename baite. Tačiau, jeigu nutinka taip, kad programos eigoje telpa ne visa steganografija, programa apie tai išveda pranešimą. Bendriems LSB ir LSBH metodų testavimams pranešimų ilgiai buvo lygus LSB metodu perskaitomų baitų ilgiui tam, kad būtų galima tinkamai patikrinti metodų kokybę. Rezultate užslėptas stegotekstas yra nepastebimas žmogaus akimi (išskyrus LSBH+patobulinimas variantą).

Lyginant su LSB metodu, realizuojant antrą steganografijos metodą LSBH buvo susidurta su viena problema: sunku iš anksto nustatyti tinkamą stegoteksto ilgį. Kadangi LSB metodas skaito po vieną paveiksliuko baitą lygindamas paskutinį bitą su stegoteksto bitais ir pakeičia, jeigu jie nesutampa, galima teigti, kad LSB metodui galima pateikti paveiksliuko dydžio stegotekstą.

Tačiau antrojo metodo eigoje ne kiekvienas baitas yra pakeičiamas stegoteksto bitu, o išpildžius vieną iš sąlygų, gali būti interpretuojami iš karto du stegoteksto bitai.

4. Stegoanalizės metodų apžvalga

Kadangi paveiksliukų, kuriose slepiama informacija, yra daug ir jų struktūra yra skirtinga, tam buvo sukurta daugybė stegoanalizės metodų, kurie dalijami į:

- metodus, skirtus pastebėti slepiamą informaciją, pagal tam tikrus steganografijos algoritmus;
- „aklieji“ atradimo metodai ([Rgi11]).

Metodai, kurie skirti tam tikriems steganografijos algoritmams yra žymiai geresni ir jų stegoteksto atradimo tikimybė yra aukštesnė. Tačiau jie turi trūkumą: jeigu nežymiai pakeistume steganografijos algoritmą, tai jam skirtas stegoanalizės metodas stegoteksto neatras. Tokiu būdu „aklieji“ stegoanalizės metodai kartais yra efektyvesni ([Rgi11]).

Taipogi, stegoanalizės metodai skirstomi į pasyvius ir aktyvius metodus. Pasyvūs metodai atranda stegotekstą su tam tikra tikimybe, arba atpažįsta algoritmą, kuriuo buvo slepiamas pranešimas. Aktyvūs metodai nustato slepiamo pranešimo ilgį, jo poziciją, tam tikrus algoritmo įterpimo parametrus bei išima slepiamą informaciją ([Rgi11]).

Pagal paieškos objektą failuose, statistinius stegoanalizės metodus galima išskirstyti į ženklinius ir tikimybinius. Ženkliniai – tai metodai, kurie sukurti kodų fragmentų paieškai failuose, kitaip tariant "pirštų antspaudų", kurie lieka po steganografijos programos darbo. Tikimybinių metodų pagrindas yra tikimybės rodiklių analizė, skirta slaptiems pranešimams failuose (stegotekstams) ([Rgi11]).

Prie ženklinių metodų galima priskirti vizualias atakas, kai po stegoteksto įterpimo žmogus vizualiai gali pamatyti skirtumus, bei nereikalingos informacijos paieška. Nereikalinga informacija - tai papildomi paveiksliuko baitai, kurie įrašomi į paveiksliuko galą arba į tą spalvą, kuri paveiksliuke dubliuojasi. Prie tikimybinių metodų galima priskirti Chi-kvadrato (^{2}) statistikos analizę.

Taigi, vieni is žinomiausių stegoanalizės metodų yra:

- reguliarios ir vienetinės poros (angl. Regular and Singular (RS)) ([FGD01]);
- reikšmių poros (angl. Pairs of Values (PoV)) ([Sta05]);
- vizualios atakos (angl. Visual Attacks) ([WP99]);
- universalus aklumas (angl. Universal Blind) ([FGD02]);
- diskretinė kosinuso transformacija (DCT - Discrete Cosine Transform) ([FGD02]);
- unikali "pirštų antspaudų" (Unique Fingerprints) ([FGD02]);
- ir t.t.

Vieni iš populiariausių statistinių stegoanalizės metodų yra reguliarios ir vienetinės porų bei reikšmių porų statistinės stegoanalizės metodai.

Kaip jau buvo minėta, stegoanalizė – tai mokslas, kuris atskleidžia steganografijos metodais paslėpto pranešimo faile buvimą ar nebuvimą. O pagrindinė steganografijos užduotis – paslėpti informaciją kitame faile taip, kad stegoanalizės metodai jos neatrastų. Kitaip tariant, stegoanalizės tikslas yra nustatyti slapto pranešimo egzistavimo faktą su kuo didesne patikimumo tikimybe. Kartais stegoanalizės tikslu būna ir pranešimo iššifravimas, nors tai ir nėra jo pirminis tikslas.

Vienas iš plačiausiai naudojamų metodų yra LSB, kadangi jis yra paprastesnis, nereikalaujantis sudėtingų formulių. Šiuo metodu užkoduotas pranešimas paveiksliuke yra vizualiai nepastebimas. Manoma, kad LSB metodas įdeda reikšmes pagal atsitiktinę eilę, tačiau jeigu įsigilintume į metodo veikimą, galima suprasti, kad iš tikrųjų tai nėra tiesa. Šitas metodas slepia stegoteksto bitą kiekviename konteinerio baite. Tačiau, jis pakeičia mažiausio reikšmingumo bitą, todėl vizualiai paveiksliuke nesimato pakeitimų.

4.1. Reguliari - Vienetinė stegoanalizė

Reguliari - Vienetinė stegoanalizė (Regular - Singular (RS)) – tai stegoanalizės metodas, kurį pristatė Jessica Fridrich, Rui Du ir Meng Long kolektyvo 2001 metais, Niujorke ([BB07]).

Metodo realizacija – tai dviguba statistika reguliarių ir vienetinių porų, gretutinių pikselių. Jis turi aukštą slapto pranešimo atradimo tikslumą ir leidžia apytiksliai nustatyti jo ilgį. Matematinis modelis yra nesudėtingas, dažniausiai kvadratinė lygtis.

Taigi visų pirma paveiksliukas dalinamas į n pikselių grupes $G(x_1, x_2, ..., x_n)$, kur n yra lyginis skaičius (pavyzdžiui 2) pikselių(šalia vienas kito) porų - pagal histogramos horizontalę. Tai pikselių grupei nustatoma reguliarumo arba glodumo $f(G)$ funkcija. Tokia funkcija gali būti bet kokia, tarkim reikšmių dispersija grupėje arba skirtumų suma tarp gretutinių pikselių. Pikselis tai sveikas skaičius P nuo 0 iki 255 ([FGD01]) :

$$f(x_1, x_2, ..., x_n) = \sum_{i=1}^{n-1} |x_{i+1} - x_i|,\ f(x_1, x_2, ..., x_n) \in R.$$

Funkcija $F(x)$ vadinama veidrodiniu atspindžiu (angl. *flipping*) ir turi savybę $F(F(x)) = x$, $\forall x \in P$. Suformuluotos dvi veidrodinio atspindžio funkcijos - F_1 (tai žemesniojo bito inversija) ir F_2 (inversija su perkėlimu į aukštesnįjį bitą) ([FGD01]):

F_1: $0 \leftrightarrow 1, 2 \leftrightarrow 3, \ldots, 254 \leftrightarrow 255;$

F_2: $255 \leftrightarrow 0, 1 \leftrightarrow 2, 3 \leftrightarrow 4 \ldots, 253 \leftrightarrow 254, 255 \leftrightarrow 0.$

Panaudojus F_1 ir F_2 grupei $G(x)$ gaunam pertvarkytą pikselių grupę. Toliau pertvarkytos pikselių grupės dalinamos į klases:

- reguliarios grupės: $G \in \boldsymbol{R} \Leftrightarrow f(F(G)) > f(G);$
- vienetiniai grupės: $G \in \boldsymbol{S} \Leftrightarrow f(F(G)) < f(G);$
- nenaudojamos grupės: $G \in \boldsymbol{U} \Leftrightarrow f(F(G)) = f(G).$

Ryšis tarp grupių ir paveiksliuko yra svarbiausia tyrimo dalis. Iš pradžių nustatomi grupių, patekusių į klases kiekiai R_M, S_M, U_M, R_{-M}, S_{-M}, U_{-M}, kur indeksai M ir $-M$ reiškia pasiskirstymą į F_1 ir F_{-1}. Pagrindinis tikslas yra nustatyti, kokiu būdu įterptas slaptas pranešimas, LSB metodu, gali įtakoti pikselių grupių statistiką ([FGD01]).

RS - stegoanalizės metodas sukurtas statistiniu pagrindu, nustatyti ar į paveiksliuką įterptas pranešimas. Jeigu stegoteksto nėra, turėtų būti tenkinama lygybė ([FGD01]):

$$R_M \cong R_{-M}\,,\ S_M \cong S_{-M}\,.$$

Kitaip tariant, jeigu paveiksliukas originalus, tai santykis tarp reguliarių ir vienetinių grupių turi būti artimas vienas kitam, tai yra mažai skirtis. Jeigu taip nėra, tai tikimybė, kad paveiksliuke kažkas užslėpta, yra didelė.

4.2. Reikšmių porų stegoanalizė

Andreas Westfeld ir Andreas Pfitzmann pasiūlė kitą metodą, „reikšmių poros“ (Pair of Values – PoV), kuris plačiai naudojamas JPEG paveiksliukuose ([Sta05]). Šitas metodas sukurtas statistiniu Chi-kvadrato (^{2}) požymiu lyginant gauto paveiksliuko histogramą su originalu. Metodo rezultatas – tai stegoteksto egzistavimo tikimybė konteineryje.

Reikšmių porų metodas efektyviai naudojamas stegoanalizėje nagrinėjant paveiksliukus, sukurtus LSB metodu, kai mažiausio reikšmingumo bitas pakeičiamas slepiamo pranešimo bitu. Westfeld – Pfitzmann metodas yra grindžiamas ieškant modelių ryškumo įverčių tikimybės tikruose ir paveiksliukuose su įterptu slaptu pranešimu. Pakeičiant mažiausio reikšmingumo spalvingumo komponentės bitą į slapto pranešimo bitą, ryškumo pikselis tampa lygus originalaus paveiksliuko pikselio ryškumui, arba pasikeičia vienetu su apytiksliai ½ tikimybe.

Tam kad atsekti slėpimo pėdsakus buvo sukurtas reikšmių porų analizės metodas, kuris ieško atsirandantį spalvų ryškumo dažnio dėsningumą. Tokios poros skiriasi tik mažiausio reikšmingumo bito reikšme (LSB).

Taigi, jeigu turime paveiksliuką, kurį sudaro paletė iš *256* spalvų (c_i), o tai reiškia, kad *PoVs* daugiausiai gali būti *n = 128*. Spalvų dažniai paskirstomi į dvį vektorių grupes: $X^{128 \times 1}$ ir $Y^{128 \times 1}$ tokias, kad x_k (kur $i = 2 * k$) ir y_k (kur $i = 2 * k + 1$), $0 \leq k \leq 127$. Vėliau apskaičiuojami gretutinių atspalvių vidurkiai, pagal formulę ([Sta05]):

$$z_k = \frac{(x_k + y_k)}{2}.$$

Westfeld – Pfitzmann metodas nustato minimalų spalvos dažnumą, jeigu $x_k + y_k \leq 4$, tada $x_k = y_k = z_k = 0$ ir $n = n - 1$. Kitaip tariant, jeigu *2*k* ir *2*k+1* dažnumo suma lygi arba mažesnė nei 4, tai individualios *2*k* ir *2*k+1* reikšmės yra prilyginamos nuliui ir kategorijų skaičius *n* yra sumažinamas vienetu. Chi-kvadrato (^{2}) statistika *n-1*, apskaičiuojama pagal formulę ([Sta05]):

$$\chi^2_{n-1} = \sum_{k=0}^{127} \frac{(x_k - z_k)^2}{z_k},$$

kai laisvės laipsnių skaičius yra *n-1*. Tikimasi, kad jeigu konteineris laiko slaptą pranešimą, tai χ^2_{n-1} reikšmė bus maža, nes x_k reikšmė bus arti z_k reikšmei. Atvirkščiai, jeigu konteineris neturės stegoteksto, tai χ^2_{n-1} reikšmė bus didelė, nes x_k reikšmė bus didesnė už z_k reikšmę ([Sta05]).

Paskutinis žingsnis – apskaičiuoti *p* reikšmė, kuri parodo stegoteksto egzistavimo tikimybę ([Sta05]):

$$p = 1 - \frac{1}{2^{\frac{n-1}{2}} \Gamma(\frac{n-1}{2})} \int_0^{\chi^2_{n-1}} e^{-\frac{x}{2}} \chi^{\frac{n-1}{2}-1} dx.$$

Jeigu χ^2_{n-1} reikšmė didelė, tai *p* bus lygus nuliui, t.y., stegoteksto nėra.

Pasak Westfeld ir Pfitzmann, jeigu mažiau nei 100% pikselių turi įterptą informaciją, tuomet stegoteksto egzistavimo tikimybė ženkliai sumažėja, kuomet patikrinamas didesnis procentas pikselių ([Sta05]).

5. Stegoanalizės metodų realizacija bei rezultatai

Darbo tikslas palyginti dviejų stegoanalizės metodų kokybę (kuris metodas geriau atranda slaptą pranešimą konteineryje: PoV ir RS. Abu metodai buvo realizuoti JAVA programavimo kalba. Programa skirta atpažinti stegoteksto buvimo tikimybę spalvotame BMP formato paveiksliuke. Prieš testuojant tuos metodus reikėjo surasti tinkamus duomenis. Jie buvo sukurti naudojant LSB ir LSBH metodus.

Stegoanalizės metodų analizei buvo parinkta virš 300 spalvotų 600 x 800 pikselių paveiksliukų. Paveiksliukuose LSB ir LSBH metodais buvo paslėpti skirtingų dydžių (nuo ketvirtadalies iki pilno paveiksliuko dydžio) stegotekstai. Kiek pakeistų bitų laiko kiekvienas paveiksliukas, šitoje stadijoje mums nėra svarbu, todėl jeigu paveiksliuką sudaro 10 000 baitų, tai 100 % stegoteksto ilgis bus 10 000 bitų).

Buvo pastebėta, kad naudojant tekstą ar kitą paveiksliuką kaip stegotekstą, yra pakeičiamas mažas paveiksliuko procentas. Dėl to buvo pasirinkta slėpti skirtingų ilgių atsitiktinę bitų seka. Tokiu būdu yra lengviau kontruoliuojamas paveiksliuko pakeistų baitų kiekis.

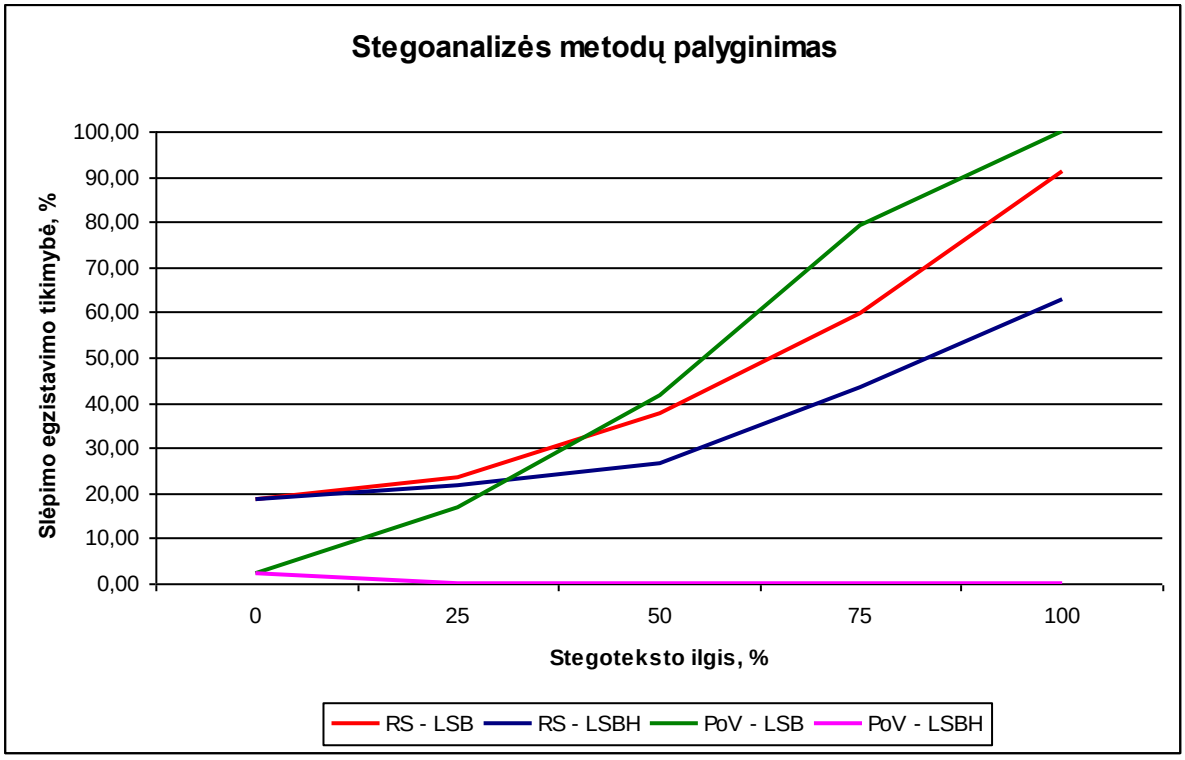
Pradžioje programai buvo paduodami originalūs paveiksliukai tam, kad įsitikinti, jog programa teisingai priima įvestus duomenis. Vėliau buvo dirbama su paveiksliukais, į kuriuos buvo įterpiami 25 %, 50 %, 75 % ir 100 % ilgio stegotekstai (5.1. lentelė). Kaip teorijoje buvo pateikta, RS stegoanalizės metodas gerai tinka histograminiams steganografijos metodams, nes jis atranda stegotekstą paveiksliuke su didesne tikimybe. Priklausomai nuo stegoteksto ilgio, tikimybė, kad RS metodas tai pastebės yra didelė − net iki 91 %. Tačiau, kartais metodas klysta ir net iki 19 % originalių paveiksliukų nustatė stegoteksto egzistavimą. Nors to priežastis gali būti netinkamai parinkti paveiksliukai arba per didelė paklaida.

Stegoteksto ilgis, %	Slėpimo egzistavimo tikimybė, %			
	RS		**PoV**	
	LSB	**LSBH**	**LSB**	**LSBH**
0	18,77		2,00	
25	23,33	21,67	17,00	0
50	37,67	26,33	41,67	0
75	59,67	43,33	79,33	0
100	91,00	62,67	100,00	0

5.1. lentelė

Taip pat pateiktoje diagramoje (5.2. diagrama) galima pamatyti metodų kokybę. Taigi, žiūrint į metodų rezultatus, galima padaryti išvadą, kad kuo ilgesnis tekstas, tuo didesnė tikimybė, kad abu stegoanalizės metodai, PoV ir RS, jį pastebės su gana didele tikimybe.

Darant išvadas iš slėpimo pusės, geriau slėpti stegotekstą yra LSBH metodų, nes jo dėka mažiau baitų yra iškraipstama ir tokius paveiksliukus sunkiai atpažįsta RS ir PoV metodai. Kalbant apie tuos, kurie nori atrasti slaptą informaciją – geriau naudoti RS metodą, nes jo dėka gana didelė tikimybė atrinkti tuos paveiksliukus, kurie savyje laiko slaptą pranešimą, užkoduota LSB arba LSBH metodais.



5.2. diagrama

Bendrame metodų palyginime, galima teigti, kad RS metodas yra kur kas efektyvesnis, nes PoV metodas visiškai nemato LSBH metodu paslėptų stegotekstų. Kaip buvo pastebėta teorijoje, PoV metodą reikia taikyti ne visam paveiksliukui, o tik jo daliai, nes *p* reikšmė kinta su mažėjančiu laipsniu ilgėjant pranešimo ilgiui.

Kartais norint su didesnė tikimybe atrasti steganografiją, reikia pritaikyti keletą stegoanalizės metodų. Pabandymui, keliems paveiksliukams su 50 % ilgio stegotekstais pradžioje buvo pritaikytas RS metodas. Paveiksliukai, kurie buvo aptikti dėl steganografijos buvimo buvo išimti iš bibliotekos ir likusiems buvo pritaikytas PoV metodas. Po RS metodo buvo aptikta 23,33% paveiksliukų, o po PoV metodo dar 4% paveiksliukų.

Taigi teorija, kad geriau taikyti keletą stegoanalizės metodų pasitvirtina.

5.1. „Reguliari-Vienetinė“(RS) stegoanalizės metodo realizavimo rezultatai

Sukurtos programos aprašymas: programai paduodama spalvotų, BMP formato, paveiksliukų biblioteka. Biblioteką sudaro paveiksliukų rinkiniai su „švariais“ paveiksliukais, t.y. su paveiksliukais, kurie neturi steganografijos viduje, ir paveiksliukai su skirtingo ilgio steganografijomis. Steganografijos tipas gali būti tekstas, nespalvotas BMP formato paveiksliukas arba atsitiktinė bitų seka. Tam, kad būtų patogiau lyginti metodus bei būtų iškraipoma daugiau bitų, stegoteksto tipu buvo išrinkta atsitiktinė bitų seka.

Programa apskaičiuoja visas RS metodo grupes ir palygina R ir S reikšmes. Tam, kad programiškai nustatyti ar tos reikšmės yra apytiksliai lygios, reikšmių santykiai yra sudedami naudojant 2% paklaidą.

Pateikiant rezultatą, programa išveda atsakymą: „Paveiksliuke užslėptas stegotekstas“ arba „Paveiksliukas neturi stegoteksto“.

5.2. „Reikšmių poros“(PoV) stegoanalizės metodo realizavimo rezultatai

Sukurtos programos aprašymas: programai paduodama spalvotų BMP formato paveiksliukų biblioteka (tokia pat kaip ir RS metodui).

Programa apskaičiuoja Chi-kvadrato (^{2}) reikšmę, *n* reikšmes ir jas įstato į tikimybę *p* išreiškiančią formulę. Tačiau buvo susidurta su problema, kad nevisada teisingai yra apskaičiuojamas integralas JAVA klasėje, todėl patikrinimui visos *p* reikšmės dar kartą buvo perskaičiuojamos MAPLE programoje.

Šitam metodui taip pat buvo pasirinkta paklaida, 20 %. Tai yra, programa apskaičiuoja Chi-kvadrato (^{2}), įstato į *p* apskaičiavimo formulę. Jeigu programa išveda $p > 0,2$, tai laikoma, kad stegotekstas egzistuoja.

Realizuojant PoV stegoanalizės metodą, buvo pastebėta, kad maža χ^2_{n-1} reikšmė gaunama tuomet, kai ji yra apitiksliai lygi laisvės laipsnių skaičiui *n-1*. Pavyzdžiui, paėmus paveiksliuką 600 x 800 be stegoteksto, jo χ^2_{n-1} reikšmė bus apie 284,82, o įterpus į jį 25% ilgio stegotekstą – 237,47, įterpus 50 % - 206,66, įterpus 100 % - 62,88.

Pateikiant rezultatą gaunamas paveiksliukų skaičius su steganografijos egzistavimu. Pastebėta, kad dėl ženkliai LSBH metodu iškraipytos histogramos, Chi-kvadrato (^{2}) reikšmės didėja priklausomai nuo stegoteksto ilgio, dėl ko vėliau sudėtinga aptikti slaptą pranešimą PoV metodu.

6. Išvados

- Didėjant internetiniu paslaugų ir internete atliekamų veiksmų skaičiui, neišvengiamai didėja ir duomenų saugojimo-slėpimo poreikis, todėl galima teigti, kad ekonominiu požiūriu, investicijos ir tyrimai šioje srityje yra naudingi ir pelningi.
- Steganogramos yra efektyvesnis duomenų saugojimo būdas, kadangi priešingai nei kriptografija, yra paslepiami patys duomenis, o ne šifruojami, todėl steganogramas gali aptikti tik specialistai - stegoanalitikai. Taip sumažinamas potencialių asmenų, kurie gali aptikti slepiamą informaciją, skaičius, o tuo pačiu ir slaptos informacijos nutekėjimo tikimybė.
- Kuo didesnis skirtingų steganografijos metodų skaičius, tuo didesnis stegoanalizės metodų skaičius, kadangi kiekvienas stegoanalizės metodas turi savo formules.
- Panaudojant keletą stegoanalizės metodu vienam steganografijos metodui, tikimybė aptikti stegotekstą padidėja.
- Didžiausias steganografijos efektyvumas yra pasiekiamas naudojant nespalvotus, monotoniškus vaizdus, turinčius minimalų kiekį piešinio elementų.
- Norint sumažinti stegoteksto aptikimo tikimybę, reikia naudoti tokį metodą, kuriam būtų galima pritaikyti kuo mažiau stegoanalizės metodų.
- Stegoanalizės metodo efektyvumas priklauso nuo jo universalumo, tai yra, kiek skirtingų steganografijos metodų gali būti analizuojami.
- Iš aprašytų stegonalizės metodų, geresniu laikomas reguliarios–vienetinės metodas, kadangi jo efektyvumas mažiau jautrus įvedimo parametrams, naudoja mažiau kompiuterio resursų, beto pasižymi aukštu jautrumu.
- RS metodas pranašesnis uz PoV, nes slaptą pranešimą galima aptikti nuskaičius vos kelis modifikuotus paveiksliuko bitus.
- LSBH metodu užslėptas pranešimas yra sunkiau aptinkamas PoV metodu; kuo ilgesnis stegotekstas, tuo labiau iškraipoma histograma ir PoV metodas sunkiai aptinka jį.
- Steganografijos metodams, kurie naudoja histogramas, geriau naudoti RS metodą.
- PoV metodą geriau taikyti po RS metodo, arba su kitais steganografijos metodais, išskyrus LSB ir LSBH.
- Nepriklausomai nuo pasirinkto steganografijos metodo, jaučiama tendencija, kad aptikimo tikimybė proporcinga stegoteksto ilgiui, todėl norint efektyviai perduoti

slaptą informaciją, yra naudingiau ją skaidyti ir perduoti naudojant keletą konteinerių, nei siųsti visą pranešimą naudojant vieną konteinerį.

- Kadangi LSBH+patobulinimas metodas turi didesnius vizualinius iškraipymus, jo efektyvumą galima padidint naudojant mažo formato paveiksliukus ir trumpus stegotekstus; tokiu būdu sumažėja vizualinis originalo ir pakeisto paveiksliuko skirtumų įvertinimas.
- Kadangi dalis stegoanalizės metodų paremtos steganografijos algoritmų ieškojimu, siekiant sukurti naują steganografijos metodą, nebūtina kurti iš naujo, o galima modifikuoti esantį.

Literatūros sąrašas

[Bat08] Philip Bateman. *Image Steganography and Steganalysis.* Department of Computing Faculty of Engineering and Physical Sciences, University of Surrey, 2008.

[Boh09] Dr. Rainer Bohme. *Advanced Statistical Steganalysis.* Springer, Germany, 2009.

[Dam03] Agnius Dambrauskas. *Kompiuterių architektūra. Atmintis.* URL:http://techno.su.lt/~IT0/dambrauskas/teorija/Kompiuteriu%20architektura/Atmintis.pdf 1,36 MB, 2003.

[Jud01] James C. Judge. *Steganography: Past, Present.Future.* URL:http://www.sans.org/reading_room/whitepapers/stenganography-/steganography-past-present-future_552, 572 KB, 2001.

[Fed12] Jekaterina Fedina. *Statistiniai stegoanalizės metodai.* Mokslo tiriamojo darbo projektas, VU MIF, 2012.

[FGD01] J. Fridrich, M. Goljan, R.Du.*Reliable Detection of LSB Steganography in Color and Grayscale Images.* Binghamtonas, Niujorkas, SUNY, 2001.

[FGD02] J. Fridrich, M. Goljan. *Practical Steganalysis of Digital Images – State of the Art.* Binghamtonas, Niujorkas, SUNY, 2002.

[Gre09] Tim Greene. *The history of steganography.* URL: http://www.networkworld.com/slideshows/2009/090809-steganography.html#slide2, 121 KB, 2009.

[MB10] Audronė Mikalauskienė, Zenonas Brazaitis. *Informacinių sistemų sauga*. VU leidykla, Vilnius, 2010.

[PA03] Programming Assignment. *Hamming Codes in TOY.*

URL:http://www.cs.princeton.edu/courses/archive/fall03/cs126/assignments/hamming.html 8 KB, 2003

[Pra06] Gujar Sujit Prakash. *Measures for Classification and Detection in Steganalysis*. Computer Science and Automation Indian Institute of Science. Bangalore, 2006.

[Rgi11] Grigory. *Steganogrfaja i stegoanaliz ot A do Ja*.
URL: http://gr1g0ry.blogspot.com/ 125 KB, 2011.

[Sta05] Christy A.Stanley. *Pairs of Values and the Chi-squared Attack.*
Department of Mathematics, Ajova Valstijos Universitetas, JAV, 2005.

[Sta06] Vilius Stakėnas. *Kodai ir šifrai.*
URL: http://www.mif.vu.lt/lmd/kodai_sifrai.pdf 2MB, 2006.

[Wiki11a] *Steganography*.
URL: http://en.wikipedia.org/wiki/Steganography 130 KB, 2011.

[Wiki11b] *Steganalysis*.
URL: http://en.wikipedia.org/wiki/Steganalysis, 38 KB, 2011.

[WP99] Andreas Westfeld and Andreas Pfitzmann. *Attacks on Steganographic Systems. Proc. Information Hiding—3rd Int'l Workshop*, Springer Verlag, 1999.