KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

INFORMACIJOS IR INFORMACINIŲ TECHNOLOGIJŲ SAUGA (6211BX008)

JONAS PETRAŠKA

Steganografinis metodas naudojantis optimizuotas kvantavimo lenteles

Baigiamasis magistro studijų projektas

Vadovas

Prof. Algimantas Venčkauskas

Kaunas, 2023

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

INFORMACIJOS IR INFORMACINIŲ TECHNOLOGIJŲ SAUGA (6211BX008)

JONAS PETRAŠKA

Steganografinis metodas naudojantis optimizuotas kvantavimo lenteles

Baigiamasis magistro studijų projektas

Vadovas

Prof. Algimantas Venčkauskas

Recenzentas

Doc. Audronė Janavičiūtė

Studentas

Jonas Petraška

Kaunas, 2023

Kauno technologijos universitetas

Informatikos fakultetas

Jonas Petraška

Steganografinis metodas naudojantis optimizuotas kvantavimo lenteles

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

1. baigiamąjį projektą parengiau savarankiškai ir sąžiningai, nepažeisdama(s) kitų asmenų autoriaus ar kitų teisių, laikydamasi(s) Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymo nuostatų, Kauno technologijos universiteto (toliau – Universitetas) intelektinės nuosavybės valdymo ir perdavimo nuostatų bei Universiteto akademinės etikos kodekse nustatytų etikos reikalavimų;

2. baigiamajame projekte visi pateikti duomenys ir tyrimų rezultatai yra teisingi ir gauti teisėtai, nei viena šio projekto dalis nėra plagijuota nuo jokių spausdintinių ar elektroninių šaltinių, visos baigiamojo projekto tekste pateiktos citatos ir nuorodos yra nurodytos literatūros sąraše;

3. įstatymų nenumatytų piniginių sumų už baigiamąjį projektą ar jo dalis niekam nesu mokėjęs (-usi);

4. suprantu, kad išaiškėjus nesąžiningumo ar kitų asmenų teisių pažeidimo faktui, man bus taikomos akademinės nuobaudos pagal Universitete galiojančią tvarką ir būsiu pašalinta(s) iš Universiteto, o baigiamasis projektas gali būti pateiktas Akademinės etikos ir procedūrų kontrolieriaus tarnybai nagrinėjant galimą akademinės etikos pažeidimą.

Jonas Petraška

Patvirtinta elektroniniu būdu

Petraška, Jonas. „Steganografinis metodas naudojantis optimizuotas kvantavimo lenteles“. Magistro studijų baigiamasis projektas / vadovas prof. Algimantas Venčkauskas; Kauno technologijos universitetas, Informatikos fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Kompiuterių mokslai, Informacijos ir informacinių technologijų sauga

Reikšminiai žodžiai: Steganografija, F5, autorinių teisių apsauga, optimizuotos kvantavimo lentelės, skaitmeninės medijos apsauga, steganografinė talpa, F5A

Kaunas, 2023. 95 p.

Santrauka

Šiame darbe yra apžvelgiama steganografija bei jos metodai, taikomi slaptos informacijos slėpimui teksto, garso ir paveikslėlių formato failuose, taip pat skaitmeniniai vandens ženklai bei jų panaudojimo būdai skaitmeninės medijos apsaugai. Didelis dėmesys darbe yra skiriamas skaitmeninės medijos turinio autorinių teisių apsaugai. Aptariamos dažniausiai pasitaikančios skaitmeninės medijos apsaugos problemos bei jų sprendimo būdai. Taip pat pateikiamos įvairios esamos skaitmeninės medijos apsaugos priemonės, apžvelgiami jų privalumai bei trūkumai. Galiausiai pateikiamos įžvalgos apie steganografijos ir vandens ženklų panaudojimą skaitmeninės medijos apsaugai bei iškeliami tolimesni šio darbo tikslai.

Pirmoje dokumento dalyje yra atliekama skaitmeninės medijos saugos problemų analizė bei apžvelgiami jų sprendimo būdai. Nagrinėjama kaip esami skaitmeninės medijos apsaugos būdai naudojami šiais laikais bei kokios problemos iškyla taikant šiuos metodus.

Antroje dalyje išsikeliami pagrindiniai reikalavimai rastoms skaitmeninės medijos apsaugos problemoms spręsti ir pasiūlomas naujas steganografinis algoritmas leidžiantis išspręsti šias problemas - F5A. Sprendimas yra detalizuojamas, papasakojami jo veikimo principai. Nurodomi metodo pliusai ir minusai bei palyginimai su esamu steganografiniu metodu F5.

Trečioje dalyje aprašoma sprendimo realizacija, detaliai aprašomi naujo steganografinio algoritmo F5A žingsniai ir detaliai aprašomas kiekvieno žingsnio veikimas.

Ketvirtoje dalyje aprašomas atliktas tyrimas lyginant F5 ir F5A algoritmus, pagal MSE, PSNR bei kitus rodiklius. Metodai lyginami pagal talpos, paveikslėlio kokybės bei kitus aspektus.

Petraška, Jonas. "Steganographic Method Using Optimized Quantization Tables". Master's thesis project / supervisor prof. Algimantas Venčkauskas; Kaunas University of Technology, Faculty of Informatics.

Computer Science, Information and Information Technology Security

Steganography, F5, copyright protection, optimized quantization tables, digital media protection, steganographic capacity, F5A

Kaunas, 2023. 95 p.

Summary

This paper provides an overview of steganography and its methods for hiding sensitive information in text, audio and image files, as well as digital watermarks and their applications for digital media protection. A major focus of the work is on copyright protection of digital media content. The most common problems of digital media protection and their solutions are discussed. It also presents the various existing tools for the protection of digital media and gives an overview of their advantages and disadvantages. Finally, insights into the use of steganography and watermarking for digital media protection are provided and further objectives of this work are outlined.

The first part of the paper analyses the problems of digital media security and outlines solutions. It examines how existing digital media protection techniques are used today and what problems arise in applying these techniques.

The second part sets out the main requirements for solving the digital media protection problems found and proposes a new steganographic algorithm to solve these problems, F5A. The solution is detailed and the principles of its operation are described. The pros and cons of the method are outlined and comparisons with the existing steganographic method F5 are made.

The third part describes the realisation of the solution, detailing the steps of the new steganographic algorithm F5A and the performance of each step in detail.

The fourth part describes the study carried out comparing the F5 and F5A algorithms in terms of MSE, PSNR and other metrics. The methods are compared in terms of capacity, image quality and other aspects.

Turinys

Paveikslų sąrašas

Lentelių sąrašas

Įvadas

Daugelis mūsų naudojamės įvairia skaitmenine medžiaga internete. Skaitome straipsnius, rašome žinutes, keliame nuotraukas bei dalinamės jomis populiariuose socialiniuose tinkluose. Tačiau dažnai nesusimąstome, jog visas nuotraukų, vaizdo įrašų, ar kitos skaitmeninės medijos formos turinys, kurį keliame į socialinių tinklų platformas yra mūsų intelektinė nuosavybė. Ši intelektinė nuosavybė, nuo sukūrimo momento yra apsaugota intelektinės nuosavybės įstatymu, tačiau dažnai aktyviai šios intelektinės nuosavybės ginti neskubame. Vienos iš priežasčių yra žinių, laiko ir noro stoka, ypač jei dalinamės nuotraukomis ir vaizdo įrašais tik su artimais draugais ar šeima. Tačiau net ir tokiu atveju nederėtų pamiršti, jog socialinis tinklas yra vieša erdvė ir mūsų nuotraukos bei vaizdo įrašai gali būti pasidalinti mūsų draugų, ar šeimos narių bei pasiekti kur kas platesnę auditoriją nei tikėtasi. Patekęs į piktavalių rankas jūsų kurtas turinys gali tapti finansinės žalos, paniekos, galbūt net šantažo įrankiu. Dėl to yra svarbu sekti, bei apsaugoti savo sukurtą turinį nuo besaikio platinimo bei neteisėto jo panaudojimo.

Šiame darbe yra apžvelgiama steganografija bei jos metodai, taikomi slaptai informacijai slėpti teksto, garso ir paveikslėlių formato failuose, taip pat skaitmeniniai vandens ženklai bei jų panaudojimo būdai skaitmeninės medijos apsaugai. Didelis dėmesys darbe yra skiriamas skaitmeninės medijos turinio autorinių teisių apsaugai. Aptariamos dažniausiai pasitaikančios skaitmeninės medijos apsaugos problemos bei jų sprendimo būdai. Taip pat pateikiamos įvairios esamos skaitmeninės medijos apsaugos priemonės, apžvelgiami jų privalumai bei trūkumai. Galiausiai pateikiamos įžvalgos apie steganografijos ir vandens ženklų panaudojimą skaitmeninės medijos apsaugai bei iškeliami tolimesni šio darbo tikslai. Skaitmeninės medijos apsaugos problemai spręsti kuriamas naujas steganografinis metodas, atliekamas tyrimas bei pateikiamos tyrimo metu padarytos išvados.

1. Steganografijos panaudojimo skaitmeninės medijos apsaugai analizė

1.1. Skaitmeninės medijos saugos problemos

Šiais laikais daugelis net nesusimąstome apie savo į interneto platybes keliamą turinį. Skaitmeninė medija, arba kitaip, vaizdo, garso, teksto bei kiti failai yra nuolatos dalinami, peržiūrimi bei redaguojami per daug nesusimąstant apie galimus autorinių teisių pažeidimus. Būtent šie pažeidimai yra pagrindinė skaitmeninės medijos saugos problemų dalis. Niekam ne paslaptis, jog interneto technologijos yra pakankamai pažengusios ir leidžia sukurti darbo vietas bei verslus milijonams žmonių. Dažnu atveju šie verslai bei organizacijos yra priklausomos nuo savo kuriamo turinio, nes tai tampa pagrindiniu jų pelno šaltiniu. Pagrindinė problema atsirado tuomet, kai žmonės suprato, jog skaitmeninį turinį galima lengvai kopijuoti bei juo dalintis.

Šiais laikais turinio kopijavimas bei įvairūs sprendimai susiję su nelegaliu turinio kopijavimu sukelia didelius nuostolius organizacijoms bei fiziniams asmenims. Konkrečias skaitmeninės medijos saugos problemas aptarsime tolimesniuose poskyriuose :

1.1.1. Piratavimas

Informacinių technologijų pasaulyje terminas „piratavimas“ reiškia nelegalų skaitmeninio turinio kopijavimą arba modifikavimą pažeidžiant turinio autorines teises. Pirataujant yra sukeliama žala autorinių teisių savininkams, prarandamas pelnas, kuris būtų gautas, jei asmenys būtų įsigiję produktą legaliai. Šiuolaikiniai piratavimo mastai internete yra milžiniški, pagal atliktus tyrimus [1] maždaug 3 milijardai žmonių kas mėnesį nelegaliai siunčiasi muzikinius įrašus, taip sukeldami žalą muzikos kūrėjams, mažindami jų pelną.

1.1.2. „Peer-to-peer“ (P2P) technologijos panaudojimas

P2P technologija yra tinklo modelis, kuriame keitimasis resursais vyksta tiesiogiai tarp vartotojų, priešingai nei kliento-serverio modelyje, resursai nėra sukaupiami viename centre iš kurio klientai juos galėtų pasiimti, tačiau yra gaunami ir teikiami vartotojų, vieni kitiems. Vienas žinomiausių P2P pritaikymo pavyzdžių yra įvairios torentų failų dalinimosi programos. Dažnai apie vadinamuosius torentus girdime iš įvairių naujienų tinklaraščių, kuriuose yra rašoma apie tai, kaip jie yra naudojami nelegaliam turiniui siųstis. Šis turinys būna kopijuojamas ir įkeliamas dalinimuisi su kitais žmonėmis neturint tam licencijos ar leidimo, tokiu būdu darant žalą autorinių teisių savininkams. Tačiau P2P technologija ir torentų failai iš esmės nėra nelegalūs, jei yra naudojami nepažeidžiant autorinių teisių, dalinantis savo sukurtu turiniu, tačiau ši technologija yra dažnai naudojama nelegaliems tikslams ir yra vienas pagrindinių piratavimo būdų.

[1] Neha Navneet Patil, “INTELLECTUAL PROPERTY RIGHTS: CHALLENGES OF ENFORCEMENT OF PROTECTION OF COPYRIGHT LAWS IN THE DIGITAL ERA,” *Intellect. Prop. RIGHTS*, vol. 3, no. 4, p. 12, Aug. 2020.

1.1.3. Failų įkėlimas ir dalinimasis be leidimo

Failų įkėlimas ir dalinimasis taip pat nėra nelegalus, jei naudojamas nepažeidžiant autorinių teisių, tačiau dažnai pasitaiko ir tokių atvejų kai yra kopijuojami ir platinami failai, kurių platinimui leidimas nebuvo gautas. Tokiu atveju pažeidėjas įkelia į failų įkėlimo platformą failą ir jo atsisiuntimo nuorodą platina internete. Žmonės radę šią nuorodą gali atsisiųsti nelegaliai gautus duomenis net nežinodami, jog šie duomenys buvo gauti nusikalstamu būdu. Toks failų platinimas pažeidžia skaitmeninių failų saugą, pažeidžiant jų savininkų autorines teises.

1.1.4. Vogto turinio pasisavinimas

Šiais laikais socialiniuose tinkluose platinami vaizdo įrašai susilaukia ypač daug dėmesio ne tik iš jaunesnės auditorijos, bet ir iš vyresnio amžiaus žmonių. Kylantį socialinių tinklų populiarumą pastebi ir įvairios reklamos agentūros, kurios yra suinteresuotos pasiekti savo tikslinę auditoriją bei reklamuoti savo produktus. Jos yra pasiruošusios už tai sumokėti didelį atlygį, priklausantį nuo peržiūrų skaičiaus bei kitų kriterijų ir tai paskatina turinio kūrėjus kurti labiau įtraukiantį, originalų turinį. Tačiau ne visi turinio kūrėjai elgiasi sąžiningai. Dažnai siekiant pasipelnyti socialiniuose tinkluose, kuriamos paskyros, kuriose keliamas vogtas turinys. Kopijuojant kito žmogaus turinį ir pateikiant jį kaip savo yra pažeidžiamos autorinės teisės. Tiesa yra tam sukurtų skaitmeninės medijos saugos sprendimų įgyvendintų socialinių tinklų platformose, kurios padeda atpažinti turinio dublikatą ir pranešti apie jį turinio autoriui, tačiau dauguma jų neapsaugo nuo neteisėto turinio pasisavinimo nukopijuojant įrašą iš vienos socialinių tinklų platformos ir įkeliant jį kitoje socialinių tinklų platformoje.

1.1.5. Neteisėtos nuorodos

Dalinimasis nuoroda į turinį taip pat gali būti traktuojamas kaip autorinių teisių pažeidimas. Šiuo atveju turima omenyje atvejį, kai skaitmeninės medijos turinys, tarkime nuotrauka arba vaizdo įrašas yra įterpiamas į jūsų asmeninį tinklaraštį arba socialinio tinklo paskyrą. Nors nuoroda nukreipia į tikrojo autoriaus puslapį, tačiau kito autoriaus nuotraukų arba vaizdo įrašų rodymas tretiesiems asmenims be jų sutikimo taip pat yra autorinių teisių pažeidimas. Tai yra mažesnė skaitmeninės medijos saugos problema nei piratavimas, tačiau žala sukeliama autoriui prarandant dalį pelno, nes turinį auditorija gali pasiekti ne tik jo numatytose vietose. Netgi minint autoriaus vardą ir pateikiant visas nuorodas į autoriaus socialinių tinklų paskyras būtina gauti kūrėjo sutikimą norint dalintis jo sukurtu turiniu.

1.1.6. „Fair use“ – Sąžiningo naudojimo įstatymo laužymas

Autorinėmis teisėmis apsaugoto turinio panaudojimas be savininko sutikimo yra nelegalus, tačiau tam yra išimčių. Šios išimtys yra apibrėžiamos sąžiningo naudojimo įstatyme ir leidžia tretiesiems asmenims naudoti autorinėmis teisėmis apsaugotą turinį be savininko sutikimo. Norint naudoti turinį be savininko sutikimo galime atlikdami asmeninį mokslinį darbą, kurdami turinio apžvalgą, komentuodami tam tikras turinio dalis, kaip tai yra daroma filmų, ar istorinių įvykių

apžvalgose. Taip pat naujienų reportažuose, pateikiant tik svarbiausias ištraukas iš įvykių. Kita sąžiningo naudojimo įstatyme numatoma dalis yra originalaus turinio perkūrimas savu stiliumi. Autorinės teisės nėra pažeidžiamos atliekant socialiniuose tinkluose vykdomus iššūkius, parodijuojant tam tikrą asmenį ar jo turinio siužetą. Tačiau sąžiningo naudojimo įstatyme yra nemažai neapibrėžtumų, kurie kelia problemų skaitmeninės medijos apsaugai. Turinys gali būti kopijuojamas be autoriaus žinios prisidengiant sąžiningo naudojimo įstatymu ir teisme gali būti sunku įrodyti, jog atliekami veiksmai buvo neteisėti. [2]

1.2. Skaitmeninės medijos saugos problemų sprendimo metodai

Praeitame skyrelyje apžvelgėme problemas su kuriomis žmonės susiduria bandant apsaugoti skaitmeninės medijos turinį nuo neteisėto jo panaudojimo. Visos skaitmeninės medijos saugos problemos galėtų būti atsekamos į pagrindinį kaltininką – galimybę skaitmeninius failus kopijuoti. Tokios galimybės su fiziniais objektais neturime, todėl kartais sunku tinkamai suprasti skaitmeninės medijos apsaugojimo svarbą bei pritaikyti tinkamus apsaugos sprendimus, siekiant išvengti žalos. Autorinių teisių apsauga yra sparčiai plintanti tema, dėl proporcingai sparčiai augančio turinio kiekio bei jo atnešamos finansinės naudos. Šiais laikais informacija, arba duomenys yra vienas pagrindinių internetinių išteklių, kurie gali atnešti didelę žalą arba naudą jų turėtojui. Pagrindiniai žmonės suinteresuoti skaitmeninės medijos apsauga yra įmonės bei prekių ženklai, kurių sėkmė priklauso nuo tinkamo jų intelektinės nuosavybės naudojimo.

Tolimesniuose poskyriuose aptarsime pagrindinius skaitmeninės medijos saugos problemų sprendimo būdus bei atvejus, kai sėkmingai taikomi sprendimai sugeba apsaugoti organizacijas nuo jų autorinių teisių pažeidimų.

1.2.1. Antrinė atsakomybė

Skaitmeninį turinį nėra sunku kopijuoti, tam nereikia daug laiko ar pastangų. Skaitmeniniai grafikos darbai, vaizdo, garso įrašai, taip pat programinė įranga dažnai yra kopijuojama ir platinama nelegaliai. Kiekvieno nelegalaus panaudojimo atsekti internete yra praktiškai neįmanoma, todėl teisminėje praktikoje dažnai nematome pavienių žmonių, kaltinamų piratavimu, dėl nepagrįstai didelių laiko ir finansinių sąnaudų. Vietoje to yra taikoma antrinė atsakomybė organizacijoms, kurių platformose yra dalinamasi nelegaliu turiniu. Puikus antrinės atsakomybės pritaikymo pavyzdys buvo teisminis konfliktas tarp „A & M Records" bei „Napster". „Napster" buvo garsi internetinių paslaugų platforma, kurioje naudojant P2P technologiją buvo galima dalintis failais. Ši platforma išpopuliarėjo dėl jos galimybės dalintis MP3 formato failais, kurie dažnu atveju buvo gauti nelegaliai. Įrašų kompanijos patirdavo didelę žalą, dėl neparduotų albumų, todėl padavė „Napster" į teismą. Teismo nuosprendis buvo palankus

[2] Neha Navneet Patil, "INTELLECTUAL PROPERTY RIGHTS: CHALLENGES OF ENFORCEMENT OF PROTECTION OF COPYRIGHT LAWS IN THE DIGITAL ERA," *Intellect. Prop. RIGHTS*, vol. 3, no. 4, p. 12, Aug. 2020.

įrašų kompanijai, todėl „Napster“ privalėjo savo veiklą nutraukti ir tam tikrą laiką nelegalių atsisiuntimų sumažėjo. [3]

1.2.2. Skaitmeninių teisių valdymas (DRM)

„Digital Rights Management“ arba kitaip skaitmeninių teisių valdymas yra technologija leidžianti apsaugoti failus nuo neteisėto jų panaudojimo, sekti ir apriboti turinio naudojimą bei kontroliuoti platinimą. Šios technologijos tikslas – padidinti nelegalaus panaudojimo atsakomybę, nesukeliant papildomų sunkumų vartotojui. Norint pasinaudoti skaitmeniniu turiniu vartotojas turi sutinti su terminais ir sąlygomis, nustatytais autorinių teisių savininko licencijos arba kontakto forma. Kitaip tariant, skaitmeninių teisių valdymas yra technologinis mechanizmas, leidžiantis apsaugoti skaitmeninę mediją nuo neteisėto jos panaudojimo fiksuojant kiekvieno naudotojo teisinę atsakomybę, prašant sutikti su naudojimosi terminais ir sąlygomis.

1.2.3. Kriptografinis šifravimas

Siekiant apsaugoti ypatingai svarbų turinį, nuo neteisėto jo panaudojimo gali būti taikoma ir kriptografinio šifravimo sistema. Kriptografinis šifravimas gali būti taikomas įvairių tipų failams apsaugoti. Dažnai nelegaliai parsisiuntus autorinėmis teisėmis apsaugotą turinį galime jį peržiūrėti bet kuriame kompiuteryje, tačiau taikant kriptografinį šifravimą galime apsisaugoti ir leisti failo turinį peržiūrėti tik autoriaus nustatytoje aplinkoje, kuri galėtų failo turinį atšifruoti, jei tam yra suteiktas autoriaus leidimas. Taip pat kriptografija gali būti taikoma autorinių teisių patvirtinimui naudojant kriptografinį parašą. Kriptografinis parašas gali apsaugoti skaitmeninį turinį nuo neteisėto jo panaudojimo, nes kriptografinio parašo savininkas gali įrodyti, jog turinys iš tiesų priklauso jam.

1.2.4. Turinio žymėjimas

Tikriausiai vienas paprasčiausių, tačiau laiko patikrintų būdų apsaugoti savo turinį yra turinio žymėjimas. Dažnai internete matome vaizdo įrašus ar nuotraukas su tam tikrais žymėjimais ant jų. Paveikslėliams bei vaizdo įrašams tai dažniausiai yra uždengimas tam tikru sunkiai permatomu sluoksniu su organizacijos logotipu. Toks turinio žymėjimas dažnai sukelia nepasitenkinimą žiūrovams, todėl paprastai žmonės vengia kopijuoti bei nelegaliai naudoti pažymėtą turinį. Toks būdas yra pakankamai efektyvus norint apsaugoti skaitmeninės medijos failus nuo smulkaus mąsto piratavimo, tačiau šio būdo nepakanka norint apsaugoti stambios komercinės vertės turinį, nes labiau įgudę vartotojai gali apeiti šią apsaugą naudodami įvairius grafinio dizaino įrankius žymėjimams pašalinti.

[3] Neha Navneet Patil, “INTELLECTUAL PROPERTY RIGHTS: CHALLENGES OF ENFORCEMENT OF PROTECTION OF COPYRIGHT LAWS IN THE DIGITAL ERA,” Intellect. Prop. RIGHTS, vol. 3, no. 4, p. 12, Aug. 2020.

1.2.5. „Blockchain“ - Blokų grandinės technologija

Blokų grandinės technologija leidžia įvairius skaitmeninės medijos failus pažymėti bei išsaugoti jų nuosavybės teises specialiuose kontraktuose. Šie kontraktai yra įrašomi į blokų grandinę ir yra viešai pasiekiami, tačiau negalima jų pakeisti. Blokų grandinės technologija yra puikus būdas nustatyti skaitmeninės medijos turinio autorių bei apsaugoti turinį nuo neteisėto jo pasisavinimo. Tačiau ši technologija turi ir minusų. Pasikeitus turinio autorinėms teisėms ar paaiškėjus, jog kūrinys iš tiesų pažeidžia kito kūrinio autorines teises, nebus galima panaikinti įrašo blokų grandinėje, informaciją apie atliktą turinio pasisavinimą bus laikoma blokų grandinėje.

1.2.6. Skaitmeniniai vandens ženklai

Skaitmeniniai vandens ženklai yra naudojami norint pažymėti autorinėmis teisėmis apsaugotą turinį, siekiant sekti jo naudojimą. Šis žymėjimas yra atliekamas steganografinių metodų pagalba ir yra nepastebimas žiūrovams. Nors skaitmeninių vandens ženklų naudojimas tiesiogiai nepasaugo turinio nuo nelegalaus kopijavimo, bet gali padėti įrodyti kūrinio autorines teises bei atsekti neteisėto platinimo kaltininkus. Žmonės kopijuodami skaitmeniniais vandens ženklais pažymėtą turinį negali pastebėti, jog šis turinys buvo pažymėtas, todėl tokius pažeidimus yra nesunku atsekti bei imtis įvairių teisinių veiksmų autorinėms teisėms apsaugoti.

1.3. Skaitmeninės medijos saugos esamos realizacijos

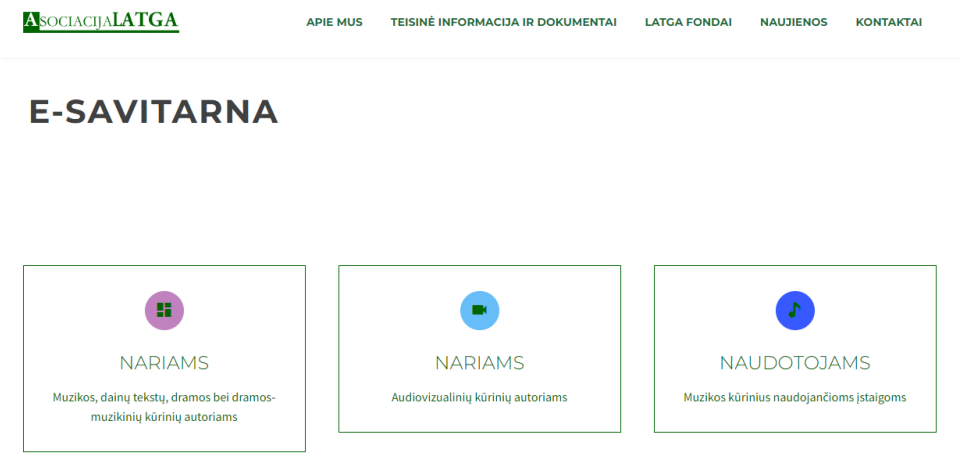
Skaitmeninės medijos saugos problemų sprendimų yra daug, tačiau ne visi šie sprendimai yra prieinami kasdieniams vartotojams. Didelės korporacijos, turinčios galimybę įgyvendinti daug pastangų bei laiko kainuojančius išskirtinius sprendimus dėl savo turinio saugumo gali jaustis ramiau, tačiau įprastiems vartotojams apsaugoti savo turinį taip pat yra galimybių. Internete galime rasti įvairių įrankių, kurie gali padėti apsaugoti savo turinį nuo galimų autorinių teisių pažeidimų.

Skaitmeninės medijos saugos sprendimų realizacijos būna įvairios. Vienos jų remiasi autorinių teisių registravimu įstaigose, steganografinių metodų, skaitmeninių vandens ženklų panaudojimu, kitos paprasta atbuline paveikslėlių paieška, ar turinio žymėjimu. Šias bei kitas esamas realizacijas apžvelgsime tolimesniuose poskyriuose :

1.3.1. LATGA

Asociacija LATGA yra ne pelno siekianti nevalstybinė organizacija, kolektyviai administruojanti įvairių sričių autorių (muzikos, literatūros, audiovizualinių, vizualiųjų ir dramos) teisių turėtojų teises, renkanti ir paskirstanti autorinį atlyginimą už kūrinių panaudojimą. Taip pat ši organizacija koordinuoja asociacijos narių veiklą, užtikrina kuo platesnį Lietuvos ir užsienio autorių kūrinių panaudojimą, užkerta kelią autorių teisių pažeidimams, finansuoja meno ir kultūros projektus, aktyviai edukuoja visuomenę autorių teisių srityje.

Naudojantis asociacijos e. savitarnos puslapiu galima registruoti savo sukurto fizinio skaitmeninio turinio autorines teises, taip apsaugant jį nuo neteisėto panaudojimo bei teikti prašymus gauti autoriaus leidimui naudoti jo turinį.

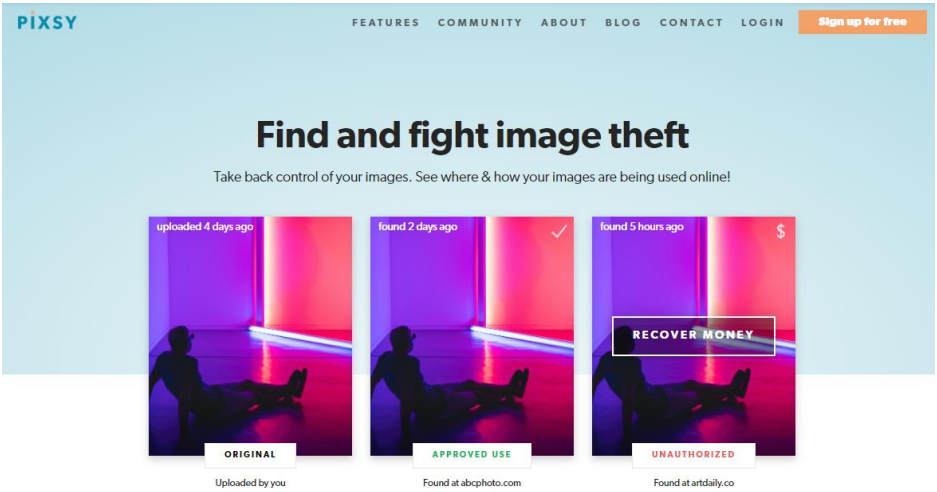


1 pav. Asociacijos LATGA e. savitarnos internetinis puslapis

1.3.2. „Pixsy“

„Pixsy“ yra internetinė platforma leidžianti naudotojams kelti savo turinį ir sekti jo panaudojimą internete. Naudodama atbulinę paveikslėlių paiešką ir dirbtinio intelekto rūšiavimo sugebėjimus, sistema nuolat tikrina, ar nėra neteisėto vaizdo naudojimo atvejų internete. Galimi pažeidimo atvejai yra suskirstyti pagal šalį ir kitus rodiklius, todėl pažeidėjus galima rasti bei rūšiuoti pagal kategorijas.

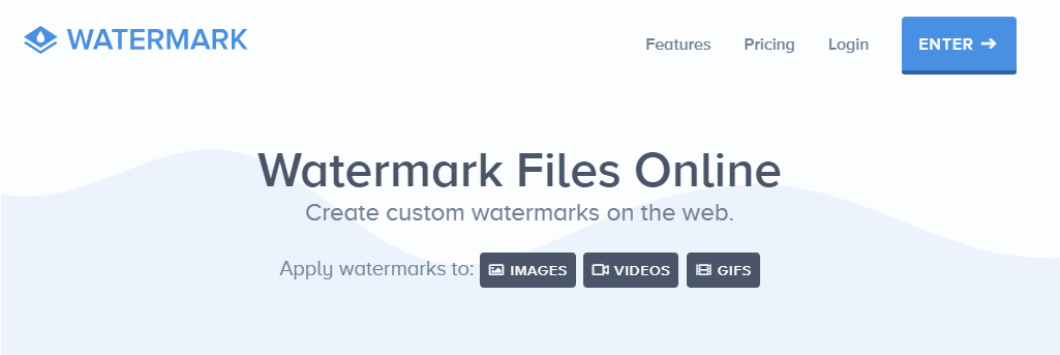
„Pixsy“ taip pat gali siųsti teisinius reikalavimus atitinkančius panaikinimo pranešimus daugiau nei 35 jurisdikcijose (įskaitant JAV, ES, Japoniją, Kiniją, Rusiją ir Indiją) naudojant daugiau nei 15 kalbų. „Pixsy“ yra integruota su daugybe vaizdų ir saugojimo debesyje paslaugų, todėl galite greitai importuoti vaizdus į savo paskyrą. Nustačius pažeidimo atvejį, „Pixsy“ sprendžia visus teisinius ir žalos padengimo aspektus. Mokama yra tik tuo atveju, jei jūsų byla bus sėkminga ir autorinės teisės yra sėkmingai apginamos.



2 pav. „Pixsy“ autorinių teisių apsaugos platforma

1.3.3. „Watermark.ws“

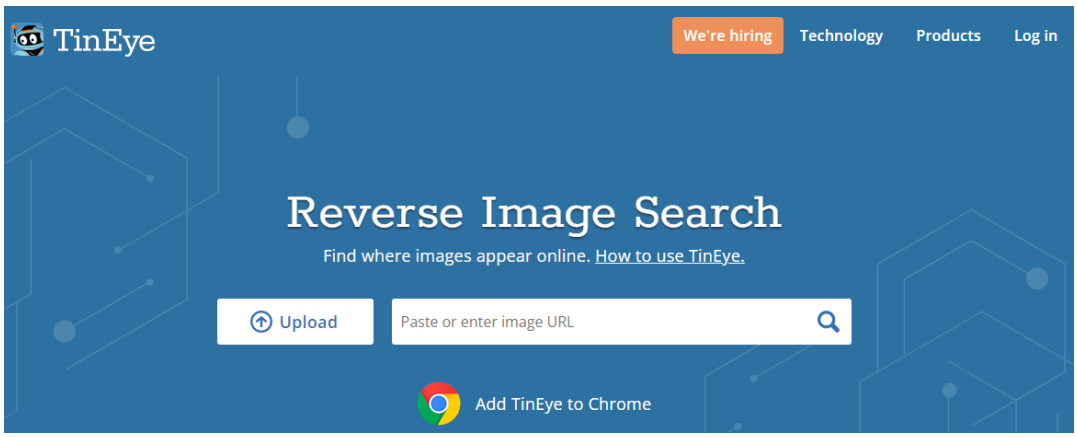
Paprastas, tačiau patikimas būdas apsaugoti skaitmeninės medijos turinį, nuo neteisėto jo panaudojimo yra turinio žymėjimas. Tokiam žymėjimui paprastai yra pasitelkiami ženklai su organizacijos logotipu. Yra sukurta daugybė panašių įrankių, galinčių uždėti ženklą ant paveikslėlio, tačiau vienas populiariausių yra „Watermark.ws“.



3 pav. „Watermark.ws“ turinio žymėjimo platforma

1.3.4. „Tineye“

„Tineye“ internetinis įrankis yra skirtas ne teisiniam žalos atlyginimo išieškojimui dėl autorinių teisių pažeidimo, bet daugiau pasitikrinimui, ar mūsų sukurtas turinys yra naudojamas be mūsų žinios. Šis įrankis yra gana paprastas, išsiskiria tuo, jog gali būti naudojamas kaip naršyklės įskiepis. Yra pakankamai paprastas ir nereikalauja didelių teisinių įsipareigojimų, siekiant išsiaiškinti autorinių teisių pažeidimo dydį.



4 pav. „Tineye“ autorinių teisių pažeidimų aptikimo įrankis

1.4. Skaitmeninių vaizdų steganografija ir vandens ženklai, jų vertinimo kriterijai

1.4.1. Steganografija

Steganografija – tai būdas paslėpti slaptą informaciją plika akimi matomoje vietoje. Slepiama informacija gali būti praktiškai bet kokie duomenys, o slepiamos informacijos kiekis priklauso nuo to kur šią informaciją norėsime paslėpti. Bene dažniausias iš Steganografijos pavyzdžių, kuriuos galime pastebėti daugelyje paveikslėlių, gali būti užkoduotas slaptas pranešimas, nurodantis tikrąjį paveikslėlio autorių arba kitą informaciją, nebūtinai susijusią su paveikslėliu. Nuslėpti informaciją, apgaunant paveikslėlio, ar kito failo stebėtoją yra pagrindinis steganografijos tikslas. Tokiu būdu nuslepiant informaciją, tampa įmanoma privati komunikacija tarp asmenų, kurie žino kokiu būdu buvo paslėptos žinutės bei kaip jas perskaityti. Steganografija nėra kriptografijos forma, joje nenaudojami raktai ar duomenų šifravimas, tačiau informacijai slėpti yra pasitelkiami sumanūs būdai, kurie priklauso nuo pasirinkto duomenų failo dydžio ir tipo. Kai kriptografija yra mokslas, daugiausia užtikrinantis privatumą, steganografija yra praktika, leidžianti slaptumą ir apgaulę.

Slaptai informacijai pernešti naudojami skirtingi formatai. Priklausomai nuo pasirinkto formato keičiasi galimas maksimalus užkoduojamos informacijos kiekis bei metodai, kurie bus naudojami informacijai slėpti. Tolimesniuose poskyriuose bus trumpai apžvelgiami skirtingi duomenų formatai bei steganografiniai metodai priskiriami kiekvienam duomenų formatui.

1.4.1.1. Teksto steganografija

Teksto steganografijoje informacija yra slepiama keičiant teksto formatą arba keičiant tam tikras teksto charakteristikas. Teksto formato failai, dažnai susideda ne tik iš teksto, tačiau ir specialiųjų simbolių bei meta informacijos apie failą, jo žymėjimą. Naudojantis šia informacija yra sukurti skirtingi teksto steganografijos metodai. Aptarsime keletą žinomų teksto steganografijos metodų.

1.4.1.1.1. Line-Shift metodas

Line-Shift, arba kitaip eilučių pastūmimo metodas. Šis metodas įgyvendinamas duomenis slepiant tarpuose tarp eilučių. Tekstiniame dokumente nežymiai keičiant tarpus tarp eilučių galime paslėpti informaciją. Šis būdas išsiskiria tuo, jog mums nebūtina žinoti, kaip atrodė originalus tekstinis failas, kad atpažintume paslėptą informaciją, nes paprastai tarpai tarp eilučių tekstiniame faile būna vienodi. Todėl žinant kaip buvo slepiama informacija, galime nesunkiai atkoduoti paslėptą žinutę.

1.4.1.1.2. Word-Shift metodas

Word-Shift, arba kitaip žodžių pastūmimo metodas. Panašiai kaip ir Line-Shift metodas, šis yra grįstas tarpų dydžio keitimu. Tačiau jame keičiami tarpai tarp pavienių žodžių, o ne tarp eilučių. Savaime suprantama žodžių tekste bus daugiau nei eilučių, todėl naudojant šį metodą bus galima paslėpti daugiau informacijos, tačiau šis metodas gali būti taikomas tik tekstiniams failams, kurie palaiko kintantį žodžių tarpų dydį. Dažnai faile pakeitus tarpus tarp žodžių gali pasikeisti ar pasislinkti eilutės tekste, dėl šios savybės norint atkoduoti paslėptą žinutę šis metodas reikalauja turėti originalų failą palyginimui.

1.4.1.1.3. Feature metodas

Feature metodas, arba kitaip ypatybės metodas. Šis metodas skirtingai nuo ankščiau minėtų dažniau gali būti sutinkamas rašytiniame formate. Jis pasižymi tam tikrų teksto ypatybių pakeitimu. Tai gali būti daugelis dalykų, nuo to kaip rašomos tam tikros raidės tekste iki raidžių aukščio ar pločio pasirinkimo. Žinant, kur ieškoti galime atrasti paslėptą žinutę, tačiau nieko neįtariančiam skaitytojui tekstas atrodys įprastas ir nesukels daug įtarimų. Šiam metodui atkoduoti reikalingas pradinis teksto variantas, jei nežinome kuriose vietose yra užkoduota žinutė, tačiau turint šią informaciją, ją atkoduoti nėra sunku.

1.4.1.2. Garso steganografija

Garso steganografija remiasi slaptos žinutės integracija į skaitmenizuotą audio signalą, pakeičiant jį taip, jog klausytojas nepastebėtų ryškaus skirtumo lyginant su originalu. Garso steganografija yra paplitusi garso takelio atpažinimo platformose tokiose, kaip „Shazam“. Pasirinktas garso takelis gali būti pažymimas tam tikra informacija, kuri galėtų identifikuoti kūrinio autorių, tokiu būdu apsaugant kūrinį, nuo galimo autorinių teisių pažeidimo bei neteisėto kūrinio pasisavinimo. Toliau apžvelgsime garso steganografijos metodus.

1.4.1.2.1. LSB Coding metodas

LSB Coding, arba kitaip mažiausios svarbos bito kodavimo metodas, remiasi skaitmeninio formato duomenų apdirbimo bitų formatu, pakeičiant, mažiausios svarbos bitą dalimi užslaptintos žinutės. Plačiau apie LSB tipo metodų veikimą yra aptariama vaizdo steganografijos poskyryje.

1.4.1.2.2. Phase Coding metodas

Phase Coding, arba kitaip fazės kodavimo metodas, išnaudoja žmogaus klausos silpnybę sunkiai atpažinti pasikeitimus garso fazėje. Žmogaus klausa lengviau sugeba atpažinti triukšmo signalą garso takelyje nei fazės pasikeitimą jame, todėl ši technika yra beveik nepastebima klausančiajam.

Naudojant šį metodą slapta žinutė yra užkoduojama kaip fazės pasikeitimai skaitmenizuotame audio signale, taip padarant šį steganografijos metodą praktiškai nepastebimu.

1.4.1.2.3. Spread Spectrum metodas

Spread Spectrum, arba kitaip spektro išskaidymo metodas, remiasi skaitmenizuoto audio signalo moduliacija ir yra daugiausia naudojamas telekomunikacijos kompanijose. Yra du pagrindiniai spektro išskaidymo įgyvendinimo būdai, tai tiesioginis sekos spektro išskaidymas, arba kitaip direct sequence spread spectrum (DSSS) ir dažnio šokinėjimo spektro išskaidymas, arba kitaip frequency hopping spread spectrum (FHSS). DSSS yra moduliacijos technika naudojama telekomunikacijos srityje.

Ši technika remiasi duomenų dauginimu su pusiau atsitiktine duomenų seka, siekiant gauti užslaptintą signalą, kurį vėliau būtų galima atkoduoti ir perskaityti užkoduotą slaptą informaciją. Užkoduojant signalą naudojant šią techniką gauname garsą panašų į triukšmą, tačiau panaudojus tą pačią pusiau atsitiktinę duomenų seką, galime pilnai atgaminti originalų garso takelį. Tuo tarpu FHSS technika, naudoja pusiau atsitiktinę duomenų seką perkoduoti pradiniam signalui, o ne pridėti papildomų duomenų prie jo, todėl galutiniame rezultate yra gaunamas vienodas dažnių pasiskirstymas.

1.4.1.2.4. Echo Hiding metodas

Echo hiding, arba kitaip aido slėpimo metodas, remiasi slapto žinutės įterpimu į originalų garso takelį aido formatu. Garso aidas, priklauso nuo trejų parametrų, pagrinde amplitudės, slopinimo ir paslinkimo nuo garso šaltinio. Naudojant aido slėpimo metodą, šie parametrai yra keičiami ir tokiu būdu yra užkoduojama slapta žinutė. Kad aidas nepatrauktų klausytojo dėmesio, paprastai yra stengiamasi parametrus keisti iki tam tikros ribos, kurios neperžengus pakeitimai tampa nepastebimi.

Šis metodas taip pat gali būti taikomas ir su video failais, kadangi šie susidaro iš vizualinės bei garsinės dalių.

1.4.1.3. Vaizdo steganografija

Vaizdo steganografija, viena populiariausių steganografijos rūšių. Paveikslėliuose patalpinta slapta informacija gali lengvai sklisti socialiniuose tinkluose bei forumuose, nesukeldama daug įtarimų. Didžiulis vaizdinės informacijos kiekis internete leidžia slaptoms žinutėms išlikti nepastebėtoms, įsimaišydamos tarp įprasto turinio jos gali išlikti paslėptos labai ilgą laiką. Norint paslėpti žinutę paveikslėlyje, nesukeliant pastebimų skirtumų su originalia paveikslėlio versija, galima padengti paveikslėlį triukšmu, naudoti spalvų variacijas ar kitas technikas žinutei paslėpti. Kurioje paveikslėlio vietoje išdėlioti mūsų slaptą informaciją, sukeliant mažiausiai pastebimų

skirtumų mums gali padėti nustatyti įvairūs vaizdo steganografijos metodai, kuriuos toliau trumpai apžvelgsime.

1.4.1.3.1. LSB – Least Significant Bit metodas

Least significant bit, arba kitaip mažiausios svarbos bito – populiariausias vaizdinės medžiagos steganografijos metodas. Tai vienas paprastesnių, tačiau dažnai sutinkamų metodų informacijai vaizduose paslėpti. Naudojant šį metodą į paveikslėlį įterpiama informacija naudojant mažiausiai svarbius paveikslėlio bitus, o mažiausiai svarbūs paveikslėlio bitai yra tie, kurių reikšmę pakeitus paveikslėlio stebėtojui mūsų perdirbtas vaizdas mažiausiai skiriasi nuo originalo. Kadangi šių bitų reikšmė maža, stebėtojui rodomos spalvos pakeitimai tampa nepastebimi.

Norint paslėpti informaciją paveikslėlyje naudojant LSB metodą negalima galutinio paveikslėlio suspausti, perdirbtam paveikslėliui saugoti turi būti pasirinktas suspaudimo be nuostolių formatas, kitu atveju paslėpta informacija tampa iškraipyta ir nebenuskaitoma. Paveikslėlių spalvose slepiant informaciją kiekviename pikselyje galime patalpinti 3 bitus duomenų, po vieną bitą raudonos, žalios ir mėlynos spalvos kodui. Šio kodo dydis priklauso nuo spalvų gylio, naudojant 48 bitų spalvas galime paslėpti daugiau informacijos, nesukeldami įtarimo, nei naudojant 24 bitų spalvas.

1.4.1.3.2. Masking and filtering metodas

Masking and filtering, arba kitaip maskavimo bei filtravimo metodas dažniausiai naudojamas tik 24 bitų arba juodai baltų paveikslėlių steganografijai. Maskavimo ir filtravimo metodai savo veikimo principu yra panašūs į vandens ženklus ar specifinį braižą paveikslėlyje. Šis metodas gali būti įgyvendinamas tam tikrose paveikslėlio vietose keičiant paveikslėlio ryškumą ar kontrastą. Šis metodas nuo LSB skiriasi tuo, kad padaromi pakeitimai yra atvaizduojami vizualiai, todėl yra šansas paveikslėlio stebėtojui pamatyti pakeitimus, tačiau tinkamai naudojant šį metodą, pastebėti skirtumus būtų itin sunku.

Tačiau šis metodas turi ir pranašumų lyginant su LSB, nes naudojant Masking and filtering metodą paveikslėlis tampa atsparus paveikslėlio dydžio suspaudimui, apkirpimui ar kitoms modifikacijoms. Kadangi paveikslėlio pakeitimai atliekami vizualiai ir paveikslėlis tampa atsparus suspaudimui, šis būdas labiau tinka JPEG formato failų steganografijai nei LSB būdas.

1.4.1.3.3. Transformations metodas

Transformations, arba kitaip transformacijų metodas yra labiau komplikuotas bei prieš tai minėti metodai, nes naudojasi sudėtingomis matematinėmis funkcijomis, tokiomis kaip diskrečioji kosinuso transformacija. Diskrečioji kosinuso transformacija (DST) yra naudojama JPEG suspaudimo algoritmui. Šis algoritmas gali suspausti ne tik paveikslėlį, tačiau ir mūsų sukurtą slaptą žinutę kartu, tokiu būdu paslepiant žinutės informaciją jau suspaustame faile.

1.4.2. Steganografiniai algoritmai

Steganografinio kodavimo algoritmai, arba kitaip stegosistemos, skirtos visų rūšių duomenims įterpti į skaitmeninių failų, įskaitant tekstinių failų formatus, sudarytą kodą, vaizdą ar garsą.

Steganografinis kodavimas gali būti taikomas praktiškai visiems failų formatams, tačiau vieniems taikyti steganografinį kodavimą yra praktiškiau nei kitiems. Stegosistemos skirtos įvairiems multimedijos tipo failams yra pačios populiariausios, nes multimedijos failai turi didelį įterpimo pajėgumą, dėl neišnaudotos vietos. Taip pat multimedijos tipo failai dažniausiai būna platinami internete, kas tampa puikia terpe sklisti kenkėjiškoms programoms. Taigi dauguma šios srities straipsnių yra sutelkti į JPEG tipo paveikslėlius, nors pagrindinės idėjos ir algoritmai paprastai taikomi ir kitiems formatams. JPEG failo formatas pagrįstas diskrečiąja kosinuso transformacija (DCT) iš 8 x 8 pikselių vaizdo blokų, sukuriančia 64 DCT koeficientus. Kiekvieno koeficiento mažiausios svarbos bitai (LSB) modifikuojami, kad būtų įterpti paslėpti pranešimai. Toliau aprašomi keturi populiariausi skaitmeninių vaizdų steganografinio kodavimo įrankiai.

1.4.2.1. JSteg

Jsteg pakeičia DCT koeficientų mažiausios svarbos bitus į slapto pranešimo bitus, praleidžiant tuos koeficientus, kurių reikšmės yra 0 arba 1. Failas yra skaitomas nuosekliai, o naudojamas algoritmas nepalaiko atsitiktinių bitų pasirinkimo. Steganografinis raktas naudojamas pranešimui užšifruoti prieš įterpiant. Šis įrankis taip pat turi ir komandinės eilutės sąsają, kuri vadinasi Jsteg-shell.

1.4.2.2. JPHide

Šis algoritmas taip pat pagrįstas DCT koeficientų mažiausios svarbos bitų pakeitimu, bet to nedaro nuosekliai. Vietoj to, jis naudoja fiksuotą lentelę, kad nustatytų, kuris koeficientas bus pakeistas toliau. Taip pat šiame įrankyje yra naudojamas pusiau atsitiktinių skaičių generatorius kai kuriems iš koeficientų praleisti, kur tikimybė praleisti keičiasi priklausomai nuo to, kiek bitų jau buvo įterpta ir kiek yra palikta.

1.4.2.3. OutGuess

OutGuess steganografinio kodavimo algoritmas yra dar viena JPEG stegosistema, naudojanti mažiausios svarbos bitų kodavimą DCT koeficientams. Priešingai nei du ankstesni algoritmai, OutGuess pasirenka koeficientus atsitiktinai, naudojant pusiau atsitiktinių skaičių generavimą, naudojantį vartotojo pateiktą slaptažodį.

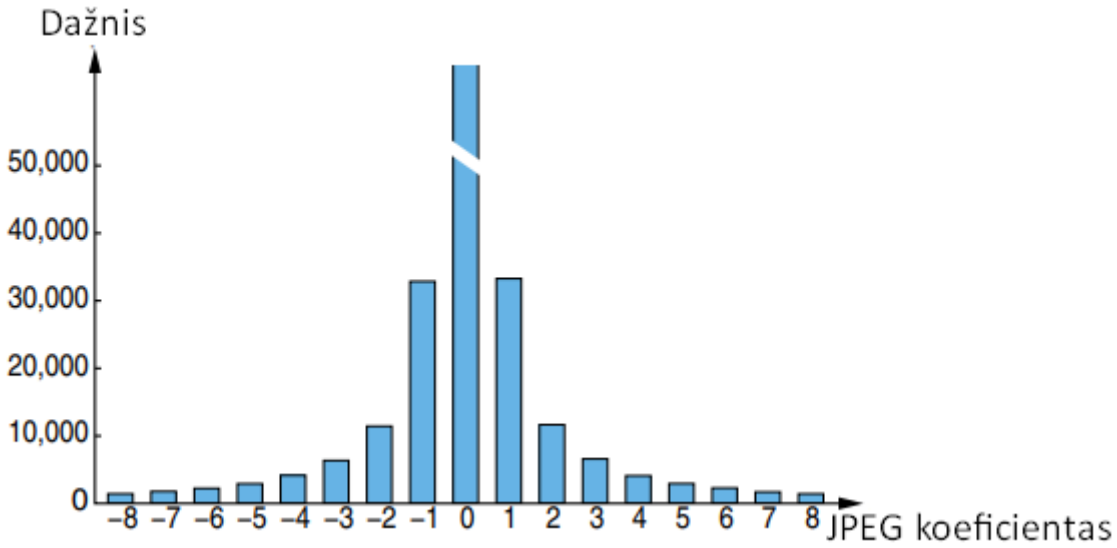
1.4.2.4. F5

Andreas‘o Westfeld‘o sukurtas F5 algoritmas gali būti vertinamas kaip aukščiau aprašytų stegosistemų tobulesnis variantas. Jame pristatoma daugybė naujų idėjų, įskaitant matricos kodavimo naudojimą, siekiant sumažinti būtinų pakeitimų skaičių. Taip pat permutacinis perjungimas, kad modifikacijos būtų tolygiai paskirstytos per visus duomenis. F5 sumažina steganografinės informacijos sklidimą per laikmeną. Dėl šios funkcijos F5 yra atsparus tam tikriems iškraipymams, pvz., dydžio keitimui ar pasukimui.

„F5“ algoritmo apžvalgą, jo autorius Andreas Westfeld‘as savo moksliniame darbe „F5—A Steganographic Algorithm High Capacity Despite Better Steganalysis“ pradeda nagrinėdamas paveikslėlių duomenų suspaudimą JPEG formato failuose. Sekant autoriaus pavyzdžiu panagrinėkime kaip yra gaunamas JPEG failas.

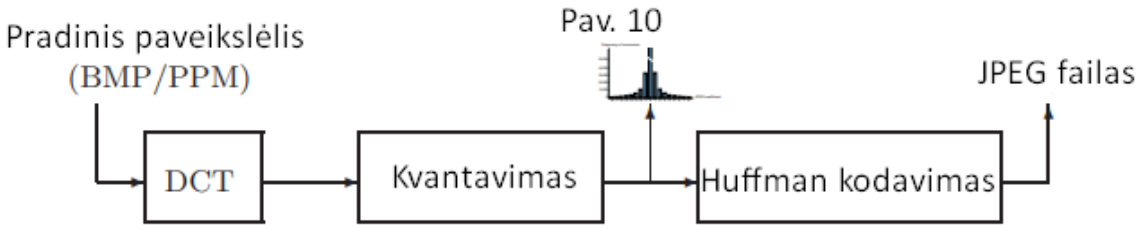
Pirmiausia JPEG kompresorius suskaido mūsų paveikslėlį į 8 x 8 dydžio matricas su pikselių reikšmėmis. Tuomet yra naudojama diskrečioji kosinuso transformacija (DCT) šioms reikšmėms transformuoti į 8 x 8 dydžio matricas su pikselių dažnio koeficientais. Po to šios reikšmės yra padalinamos iš kvantavimo matricos, kuri nurodo suspaudimo santykį (compression ratio). Šio veiksmo rezultatas yra 8 x 8 matricos su koeficientais. Žingsnyje yra prarandama dalis informacijos, todėl koeficientų matricoje atsiranda „0“ reikšmės.

Šių koeficientų dažnio pasiskirstymas gali būti naudojamas steganografinio metodo statistinei analizei, kuri parodo, ar galime aptikti failo modifikacijas.



5 pav. pavyzdinis JPEG koeficientų dažnio pasiskirstymas (Normalusis arba Dvigubas eksponentinis pasiskirstymas)

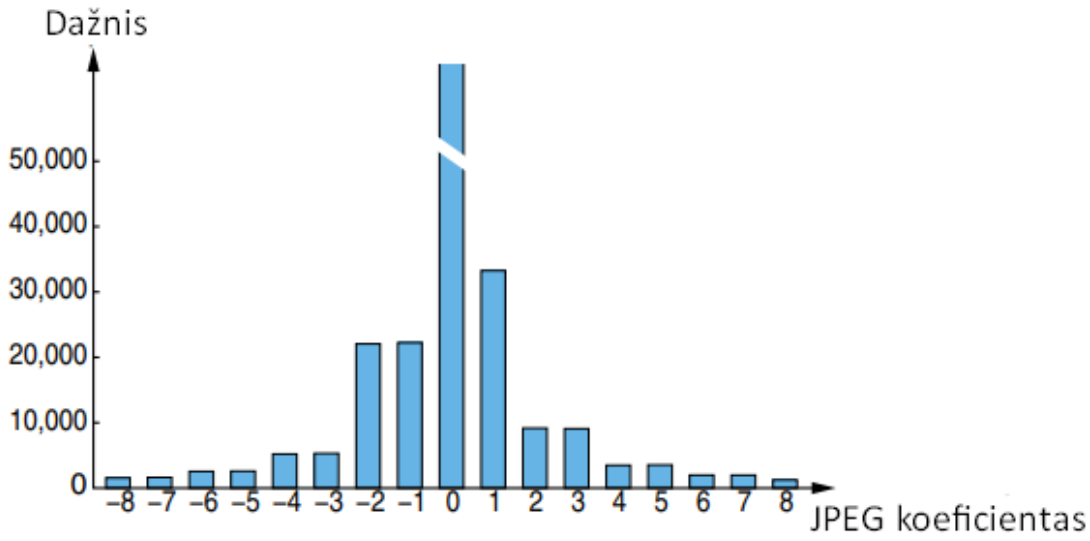
Po kvantavimo JPEG kompresijos algoritme seka Huffman‘o kodavimas, kuris pašalina kvantuotų koeficientų perteklinių reikšmių kiekį. Tačiau pagrinde nagrinėjant steganografinius algoritmus mus domina JPEG koeficientų dažnio pasiskirstymas.



6 pav. JPEG kompresijos algoritmo žingsniai

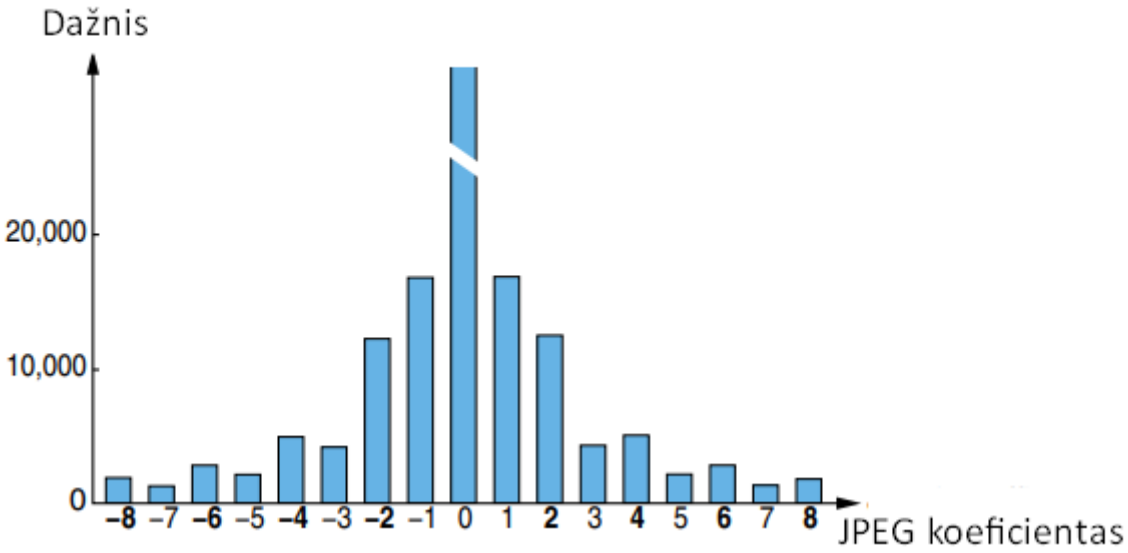
Toliau panagrinėkime kelis „F5“ algoritmo pirmtakus. Vienas jų yra algoritmas „JSTEG“.

„JSTEG“ algoritmas yra atsparus vizualioms atakoms (nepastebimas), gali paslėpti 12.8% originalaus paveikslėlio dydžio failą. Šis algoritmas remiasi JPEG kompresijos algoritmu, tačiau po kvantavimo žingsnio pakeičia koeficientų mažiausios svarbos bitus (LSB) su įterpiama (slapta) informacija. Tačiau kadangi yra keičiami bitai, koeficientų analizės metu galime pastebėti nuokrypius nuo standartinio JPEG kompresijos algoritmo, kas leidžia įtarti, jog paveikslėlyje kažkas yra paslėpta.



7 pav. JSTEG algoritmo nukrypimai nuo standartinio JPEG kompresijos koeficientų dažnio pasiskirstymo

Algoritmas „F3“ su pastebimu koeficientų dažnio pasiskirstymo nuokrypiu bando kovoti nekeisdamas koeficientų bitų reikšmių, tačiau jis mažina koeficientų reikšmes, o sumažėjimuose koduoja slaptą informaciją. Tačiau šis metodas taip pat turi problemą – kai kurios reikšmės tampa sumažintos iki „0“ reikšmės ir tampa nebeįmanoma atskirti užkoduotos reikšmės nuo nulinio koeficiento. Tokiu atveju bandoma bitą koduoti vėl, kitoje vietoje, tačiau šis pakartotinis kodavimas vėl sukuria pastebimą nuokrypį koeficientų dažnio pasiskirstymo grafike. Šiuo atveju tai pasireiškia didesniu lyginių skaičių koeficientų kiekiu grafike.



8 pav. F3 algoritmo koeficientų dažnio pasiskirstymo reikšmių nuokrypis, pasireiškiantis didesniu lyginių skaičių kiekiu

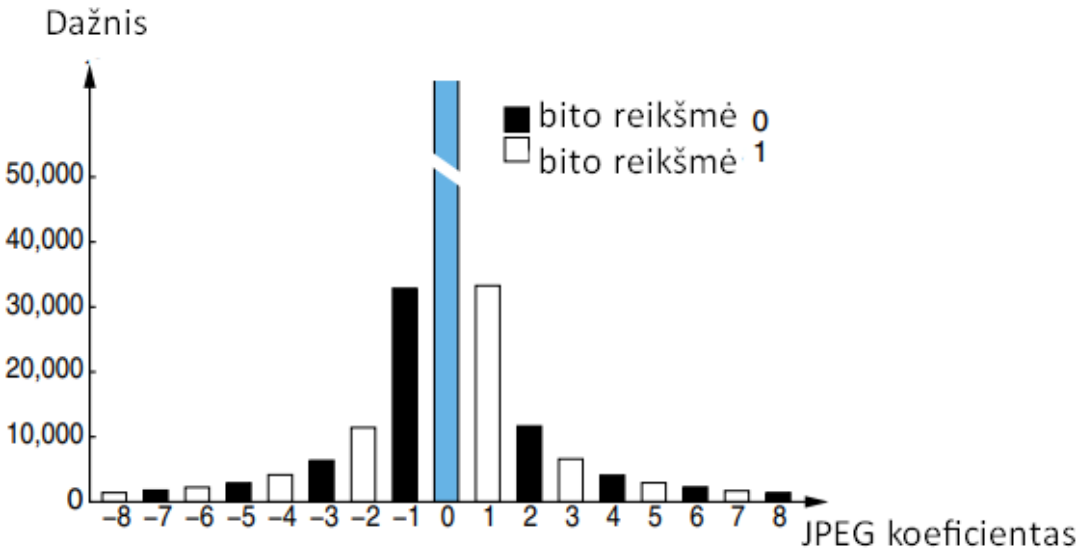
„F3“ algoritmo pagrindinė problema : pastebimas nuokrypis koeficientų dažnių pasiskirstymo grafike dėl netolygiai pasiskirsčiusių lyginių ir nelyginių koeficientų. Ši problema yra sprendžiama algoritme „F4“.

„F4“ algoritmas šią problemą sprendžia priskirdamas neigiamus koeficientus į apverstas slaptos žinutės bitų reikšmes. Vertimas vykdomas tokia tvarka :

Neigiamos reikšmės :
- Lyginės – bito reikšmė 1
- Nelyginės – bito reikšmė 0

Teigiamos reikšmės :
- Lyginės – bito reikšmė 0
- Nelyginės bito reikšmė 1

Tokiu būdu koduojant reikšmes koeficientų dažnių pasiskirstymo grafikas normalizuojamas ir nuokrypis pastebimas nėra.



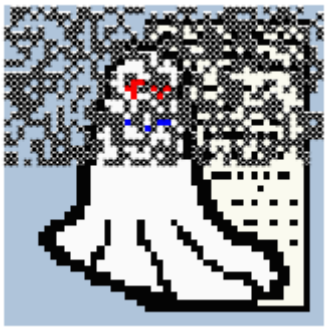
9 pav. F4 algoritmo koeficientų dažnio pasiskirstymas, koduojamos reikšmės 0 ir 1

Galiausiai, kai turime bendrą supratimą apie „F“ linijos algoritmus, galime apžvelgti kuo „F5“ algoritmas patobulino prieš jį ėjusį „F4“ algoritmą.

Dažnai slepiama informacija neužima visos galimos vietos, į kurią steganografinis algoritmas įrašo paslėptą informaciją, todėl dalis failo lieka neapdirbta ir tai gali būti pastebėta naudojant įvairias steganografines atakas. Siekiant apsisaugoti nuo tokių atvejų „F5“ algoritmas įveda naujovę - Permutatyvinį perėjimą. (Permutative straddling)

„F5“ algoritmo permutatyvinio perėjimo pagrindinė idėja yra paskleisti slepiamą informaciją per visą failą, failo vietas nustatant pagal nurodytą raktą, iš kurio vėliau galima atkurti paslėptą informaciją. Tokiu būdu steganografinis apdirbimas tampa mažiau pastebimas bei padidėja atsparumas failo modifikacijoms.

10 pav. Pradinis paveikslėlis



11 pav. "F4" algoritmas

12 pav. "F5" algoritmas

Kita „F5“ algoritme įvesta naujovė - Matricos kodavimas.

Siekiant sumažinti įterpimo operacijų kiekį „F5“ algoritme slapta informacija yra įterpiama naudojantis XOR logine operacija. Sakykime, jog paveikslėlis yra suskaitomas į dalis po 3 pikselius,

tuomet siekiant įterpti slaptą 2 bitų informaciją į paveikslėlį, mums reikia taikyti tokias XOR taisykles:

A XOR C = pirmas įterpiamas bitas.
B XOR C = antras įterpiamas bitas.

Čia A, B, C - paveikslėlio pikseliai suskaidyti į dalis po 3.

„F5“ algoritmo implementacijos žingsniai :

1. Pradėti JPEG kompresiją, sustoti po kvantavimo.
2. Sugeneruoti stiprų kriptografiškai stiprų atsitiktinių skaičių generatorių iš duoto slaptažodžio.
3. Vykdyti permutaciją (su atsitiktiniu skaičių generatoriumi ir koeficientų skaičiumi)
4. Nustatyti pradinio paveikslėlio talpą ir slepiamos žinutės ilgį.
5. Apskaičiuoti raktinio žodžio ilgį.
6. Įterpti slaptą informaciją.
7. Tęsti JPEG kompresijos algoritmą.

1.4.3. Skaitmeniniai vandens ženklai

Skaitmeniniai vandens ženklai yra būdas įterpti informaciją į skaitmeninės medijos failą, tokiu būdu, jog turinys atrodytų nepakitęs. Turinyje įprastai yra slepiama informacija apie turinio savininką, jos pobūdis gali skirtis, tačiau ji padeda identifikuoti kūrinio autorių ir apsaugoti jo autorines teises. Skaitmeninių vandens ženklų taikymas yra susijęs su steganografija. Steganografijos bei skaitmeninių vandens ženklų taikyme informacija yra paslepiama faile, tačiau skiriasi jų paskirtis. Steganografijos pagrindinis tikslas yra slaptas bendravimas, žinutes perduodant steganografiniais metodais apdirbtuose failuose, o skaitmeninių vandens ženklų pagrindinis tikslas yra autorinių teisių apsauga, tačiau abiem atvejais yra naudojami tie patys steganografinio informacijos slėpimo metodai. [4]

1.4.4. Steganografijos vertinimo kriterijai

Steganografija gali būti įgyvendinama skirtingais metodais. Tinkamo metodo pasirinkimas priklauso nuo mūsų taikomosios srities, biudžeto bei galimybių. Tačiau tinkamo metodo pasirinkimą gali palengvinti apibrėžti steganografinių metodų vertinimo kriterijai. Steganografijos vertinimo kriterijai yra tokie :

[4] K. S. Mohamed, “Data Hiding: Steganography and Watermarking,” in *New Frontiers in Cryptography*, Springer, 2020, pp. 89–98.

- Tvirtumas – steganografinio metodo atsparumas failo modifikacijoms, tokioms kaip ištampymas, apkarpymas, failo suspaudimas ar kitas apdirbimas grafinio dizaino programine įranga.

- Nepastebimumas – steganografinis metodo išpildymas turi būti nepastebimas žiūrint žmogaus akimis, tačiau kai kurie metodai papildomai bando failus paversti nepastebimais ir analizuojant failus specialiais stegoanalitiniais įrankiais.

- Bitų klaidos dažnis (BER) – bitų klaidos dažnis žymi santykį taip klaidų kiekio ir visų bitų faile kiekio. Didesnė reikšmė gali rodyti komunikacijos kanalo nepatikimumą arba steganografinio metodo neefektyvumą.

- Vidutinė kvadratinė paklaida (MSE) – ji yra apskaičiuojama lyginant baitų skirtumus tarp originalaus ir steganografiniu būdu apdirbto failo. Didesnė paklaida gali rodyti didesnį slepiamos informacijos kiekį arba metodo neefektyvumą.

- Didžiausias signalo ir triukšmo santykis (PSNR) – santykis tarp didžiausios galios signalo ir triukšmo. Didesnė reikšmę dažniausiai nurodo geresnę steganografiniais metodais apdirbto failo kokybę.

1.4.5. Skaitmeninių vandens ženklų vertinimo kriterijai

Skaitmeniniai vandens ženklai naudoja tuos pačius steganografinius metodus, siekiant įterpti autoriaus informaciją į failą, tačiau skiriasi steganografijos ir skaitmeninių vandens ženklų taikymo tikslai. Steganografija stengiasi užtikrinti slaptą komunikaciją, o skaitmeninių vandens ženklų pagrindinis tikslas yra užtikrinti kūrinio autentiškumą, bei apsaugoti kūrinio autorines teises. Kadangi skiriasi pagrindiniai metodų tikslai, naudojamų steganografinių metodų vertinimo kriterijai taip pat skirtingi. Skaitmeninių vandens ženklų vertinimo kriterijai yra tokie :

- Efektyvumas – efektyvumas yra svarbiausias skaitmeninio vandens ženklo kriterijus. Jis žymi vandens ženklo aptikimo savybę. Priešingai nei steganografijos vertinimo kriterijuose, skaitmeniniai vandens ženklai turi būti aptinkami.

- Signalo kokybė – signalo, arba kitaip originalaus failo kokybė yra svarbi skaitmeninio vandens ženklo vertinimo savybė. Steganografiniu būdu apdirbant pradinį failą, gali atsirasti skirtumų, kurie gali sukompromituoti skaitmeninio vandens ženklo kokybę. Kuo geresnė pradinio failo kokybė, tuo mažiau pastebimų skirtumų atsiranda galutiniame rezultate. Tam tikri steganografiniai metodai gali būti daugiau arba mažiau jautrūs pradinio failo dydžiui bei kokybei.

- Ženklo dydis – skaitmeninio vandens ženklo dydis turėtų būti kuo mažesnis. Tokiu būdu yra išsaugoma originalaus failo kokybė bei sumažinamas duomenų kiekis reikalingas failui talpinti ar persiųsti.

- Tvirtumas - atsparumas failo modifikacijoms, tokioms kaip ištampymas, apkarpymas, failo suspaudimas ar kitas apdirbimas grafinio dizaino programine įranga.

1.5. Esami steganografijos ir vandens ženklų panaudojimo būdai skaitmeninės medijos apsaugai

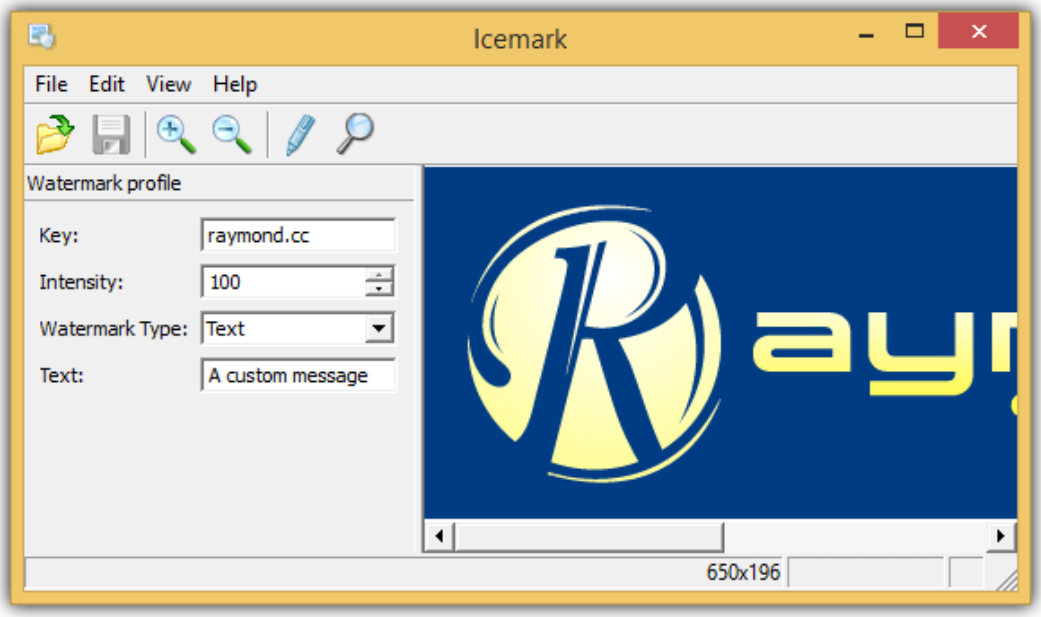
1.5.1. „SignMyImage“

„SignMyImage“ yra skaitmeninio vandens ženklo uždėjimo ant paveikslėlio įrankis, kuris taip pat turi galimybę patikrinti paveikslėlio autentiškumą. Šis įrankis pasižymi galimybe išnaudoti visus kompiuterio procesoriaus branduolius, greitesniam paveikslėlio steganografiniam apdirbimui. Su šiuo įrankiu apdirbtas paveikslėlis išlaiko skaitmeninį vandens ženklą atliekant paveikslėlio apkarpymą, suspaudimą ar dydžio keitimą. Taip pat, registruoti vartotojai gauna prieigą prie įmonės autorinių teisių apsaugos paslaugos, kuri gali pranešti apie neteisėtą jūsų paveikslėlio panaudojimą internete. Tačiau šiuo įrankiu galime pasirašyti tik 24 bitų gylio paveikslėlius. Šis įrankis nesulaukė didelio komercinio pasisekimo ir paskutinis jo atnaujinimas išleistas 2013 metais.

13 pav. „SignMyImage“ skaitmeninio vandens ženklo pridėjimo įrankis

1.5.2. „Icemark“

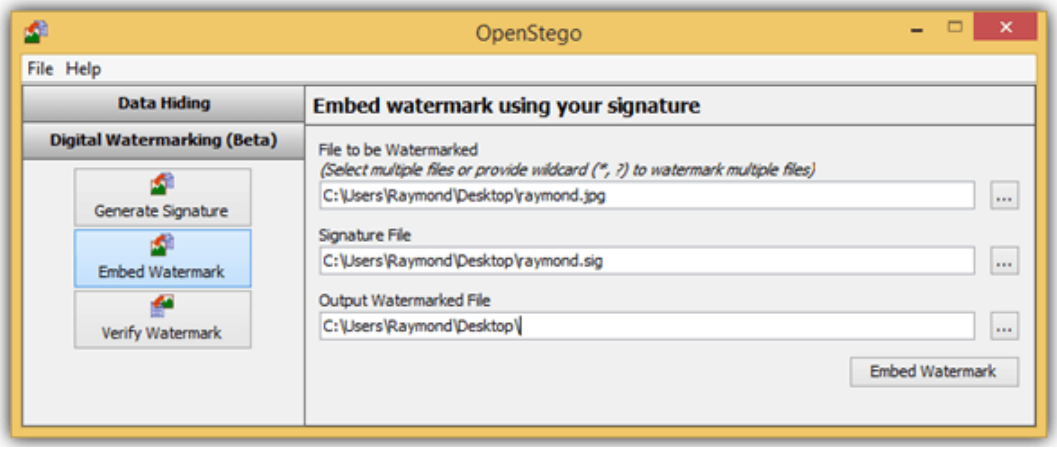
„Icemark“ yra šiek tiek naujesnis įrankis, skirtas skaitmeninių vandens ženklų pridėjimui prie paveikslėlių bei pažymėtų paveikslėlių tikrinimui bei patvirtinimui. Šio įrankio steganografiniu būdu apdirbti paveikslėliai taip pat yra atsparūs apkarpymo, suspaudimo bei dydžio keitimo operacijoms. Apribojimų failo bitų gyliui nėra. Taip pat šis įrankis leidžia suteikti paveikslėliams specialias žymes, turinio suaugusiems, apriboto naudojimo, bei neleidžiamo platinimo reikmėms.



14 pav. „Icemark“ skaitmeninių vandens ženklų pridėjimo įrankis

1.5.3. „OpenStego“

„OpenStego“ yra atviro kodo programinė įranga, skirta steganografinės informacijos slėpimui bei skaitmeninio vandens ženklo pridėjimui prie paveikslėlio, taip pat skaitmeninio vandens ženklo autentiškumo patikrinimui. Šis įrankis yra atviro kodo, todėl jo veikimas yra skaidrus ir galimi šio įrankio modifikavimai savoms reikmėms, tačiau baziniai šio įrankio pasirinkimai yra gana riboti palyginus su prieš tai aptartais programiniais įrankiais.

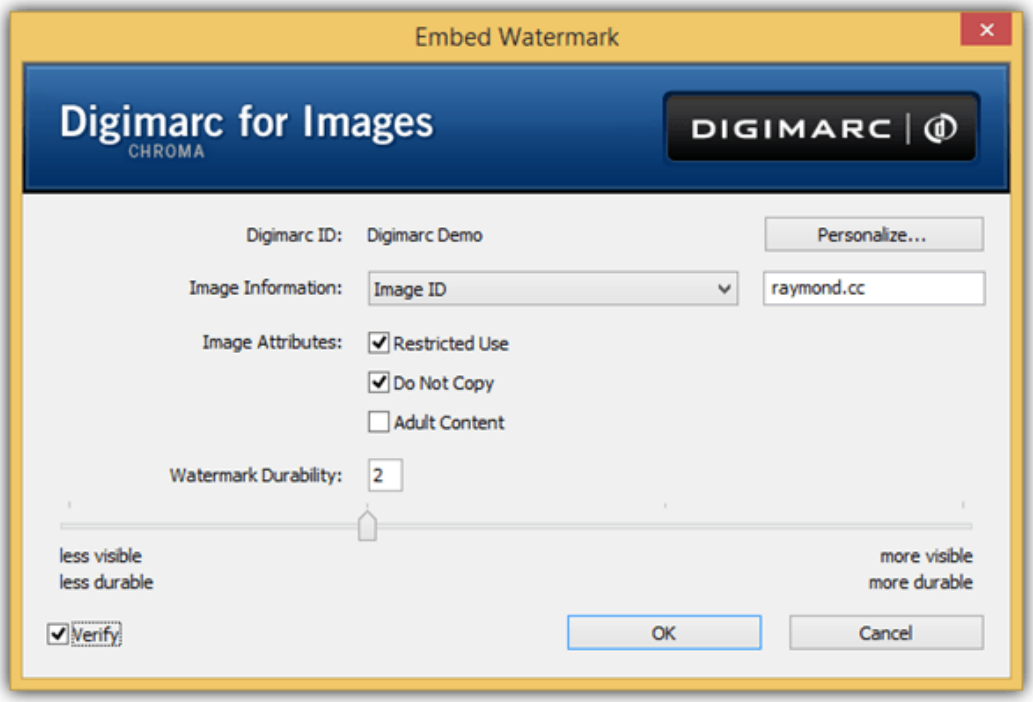


15 pav. „OpenStego“ skaitmeninio vandens ženklo pridėjimo įrankis

1.5.4. „Digimarc guardian for images“

„Digimarc guardian for images“ iš kitų skaitmeninių vandens ženklų pridėjimo įrankių išsiskiria tuo, jog šis įrankis yra naudojamas kaip įskiepis populiarioje grafinio dizaino

programinėje įrangoje „Photoshop“. Jame yra įdiegtos ir autorinių teisių apsaugos galimybės, suteikiant prieigą prie paveikslėlio autorinių teisių pažeidimų paieškos sistemos.



16 pav. „Digimarc for images“ skaitmeninio vandens ženklo pridėjimo įrankis

Išvados ir uždaviniai

1. Šioje dalyje buvo apžvelgti skirtingi steganografijos metodai skirti slaptai informacijai, teksto garso ir vaizdo formato failuose slėpti. Panagrinėtos metodų gerosios ir blogosios savybės, aptikimo tikimybės bei galimi taikymo būdai.

2. Atlikus skaitmeninės medijos saugos problemų analizę, pastebėta, jog pagrindinė visų skaitmeninės medijos saugos problemų priežastis yra lengvai suteikiama galimybė kopijuoti skaitmeninį turinį, be savininko žinios, taip pažeidžiant jo autorines teises.

3. Išanalizavus skirtingus skaitmeninės medijos saugos sprendimo metodus bei jų realizacijas, nustatyta, jog nors ir galimybių apsaugoti savo intelektinę nuosavybę yra pakankamai nemažai, praktikoje šie metodai nepadeda visiškai apsaugoti skaitmeninės medijos turinio nuo neteisėto jo naudojimo, tačiau dažniausiai suteikia tvirtą pagrindą atlyginti sukeltai žalai.

4. Steganografijos bei skaitmeninių vandens ženklų taikymas skaitmeninės medijos apsaugai yra puikus ir daug pastangų nereikalaujantis būdas apsaugoti mūsų intelektinę nuosavybę nuo neteisėto jos pasisavinimo, tačiau pastebėta, jog praktikoje didesnio populiarumo susilaukia hibridiniai sprendimai apimantys ne tik skaitmeninius vandens ženklus, tačiau ir nelegaliai įkelto turinio paieškai bei teisinių veiksmų administravimą.

5. Atlikus analizę iškelti tokie tolimesni uždaviniai :

    - Išsiaiškinti naršyklės įskiepių kūrimo ypatumus.

    - Sukurti naršyklės įskiepį, skirtą skaitmeninės medijos apsaugojimui.

    - Praktiškai realizuoti bei ištirti sukurtą įskiepį bei palyginti rezultatus su jau esamais skaitmeninės medijos apsaugos metodais.

2. Steganografinio įskiepio skirto skaitmeninės medijos apsaugai projektas

2.1. Steganografinių algoritmų panaudojimas skaitmeninei medijai apsaugoti

2.1.1. Bendrieji reikalavimai

Atlikus skaitmeninės medijos saugos problemų bei sprendimų analizę buvo nustatyta, jog pagrindinė visų skaitmeninės medijos saugos problemų priežastis yra lengvai suteikiama galimybė kopijuoti skaitmeninį turinį, be savininko žinios, taip pažeidžiant jo autorines teises. Išsiaiškintas poreikis turėti sprendimą gebantį savininkui pranešti, kuomet jo turiniu yra dalinamasi be jo sutikimo. Taip pat, apžvelgus daugelį skirtingų skaitmeninės medijos saugos sprendimų pastebėta, jog nors ir galimybių apsaugoti savo intelektinę nuosavybę yra pakankamai nemažai, praktikoje šie metodai nepadeda visiškai apsaugoti skaitmeninės medijos turinio nuo neteisėto jo naudojimo, tačiau dažniausiai suteikia tvirtą pagrindą atlyginti sukeltai žalai. Taigi kuriamas sprendimas turi ne tik atpažinti turinio autorių, tačiau ir apsaugoti jo turinį, nuo neteisėto jo panaudojimo. Didžiausio populiarumo sulaukė hibridinį modelį turintys sprendimai, kurie leidžia ne tik patikrinti tikrąjį turinio savininką, bet turi turinio paieškos, teisinio išieškojimo bei kitus turinio apsaugos būdus, todėl kuriamas sprendimas turės neapsiriboti vienu turinio apsaugos būdu, siekiant užtikrinti visapusišką autorinių teisių apsaugą.

Atsižvelgiant į šiuos pastebėjimus po analizės dalies, projektuojamam sprendimui nustatomi pagrindiniai reikalavimai :

- Atpažinti neteisėtai naudojamą turinį, ir pranešti apie tai autorinių teisių savininkui.
- Apsaugoti atpažintą turinį nuo neteisėto jo panaudojimo.

2.1.2. Sprendimo aprašymas

Bendriesiems reikalavimams patenkinti bus kuriamas autorinių teisių apsaugos sprendimas - naršyklės įskiepis gebantis atpažinti turinį pažeidžiantį autorines teises. Įskiepio efektyvumas priklausys nuo naudotojų skaičiaus, siekiant sumažinti bereikalingo naršymo kiekį įrankį įsidiegusio naudotojo kompiuteris veiks kaip autorinėmis teisėmis apsaugoto turinio paieškos sistema, kuri skanuos puslapį, kuriame vartotojas dabar yra (su galimybe išjungti, dėl saugumo sumetimų) ir radusi potencialų pažeidimą, praneštų apie tai naudotojui. Autorinių dokumentų sąrašas saugomas API, su kuriuo komunikuos įskiepis, bus siunčiami rastų paveikslėlių duomenys į API ir gaunamas atsakymas apie paveikslėlio pažeidimo statusą. Radus autorinių teisių pažeidimą autorius galės būti informuojamas dviem būdais : siunčiant laišką elektroniniu paštu arba siunčiant žinutę į jo įskiepį (įskiepis kas valandą atnaujins duomenis apie pažeidimus). Tačiau pranešimą apie autorinių teisių pažeidimą jau gali atlikti daugelis analizės dalyje minėtų sistemų, todėl būtina suteikti ir realią apsaugą nuo neteisėto turinio naudojimo.

Tai įgyvendinti yra du būdai :

- **Integracija į naršyklę, turinio blokavimas (simuliacija)**

  Integravus įskiepį į naršyklę, daugelis naudotojų galės matyti tik turinį, kuris nebuvo gautas pažeidžiant autorines teises. Integruotas įskiepis tikrintų gaunamą dokumentą ir jei jis

puslapyje yra neleidžiamas, neatitinka jo domenas su autoriaus domenu, turinys pagal pasirinkimą galės būti maskuojamas, pakeičiamas arba blokuojamas.

Kadangi naršyklės kūrėjai, tikėtina, nenorės integruoti siūlomo sprendimo, bus naudojama integracijos veikimo simuliacija.

- **Turinio apsaugojimas raktu**

  Alternatyvus būdas apsaugoti turinį yra leisti jį peržiūrėti tik pasirinktiems asmenims. Tai galėsime padaryti su įskiepio steganografine funkcija skirta paslėpti paveikslėlį kitame paveikslėlyje. Idėja yra leisti naudotojui įkelti autorinį dokumentą, tuomet apsaugoti jį maskuojant, pakeičiant kitu arba blokuojant. Steganografiniu būdu slepiant informaciją bus įtraukiamas ir raktas, be kurio paveikslėlio pamatyti nepavyks.

  Naudotojui yra gražinamas pasirinktu būdu apdirbtas paveikslėlis bei raktas su kuriuo galima pamatyti tikrąjį paveikslėlio vaizdą. Šiuo raktu autorius gali dalintis su savo auditorija.

2.1.2.1. Paveikslėlio apdirbimo būdai

Būdas „Mask“ (maskuoti) :

Tikrasis paveikslėlis yra steganografiniu būdu įterpiamas į paveikslėlį su išliejimo (blur) filtru. Steganografinio slėpimo metu įterpiant ir raktą, be kurio šio paveikslėlio pamatyti nepavyks. Tuomet apdirbtas paveikslėlis yra gražinamas naudotojui kartu su raktu. Turint raktą bei įskiepį, galima atstatyti užslėptą informaciją paveikslėlyje su filtru, tokiu būdu pateikiant originalų autoriaus turinį.

Būdas „Switch“ (pakeisti) :

Tikrasis paveikslėlis yra įterpiamas į kitą paveikslėlį. Steganografinio slėpimo metu taip pat įterpiant raktą. Gražinamas pakeistas paveikslėlis kartu su raktu. Įskiepis su raktu galės atstatyti originalų paveikslėlį puslapyje.

Būdas „Default“ (standartinis) :

Tikrasis paveikslėlis paliekamas, tačiau jame steganografiniu būdu yra paslepiama autoriaus informacija. Radus tokį paveikslėlį puslapyje jis yra siunčiamas į API patikrinimui ar nepažeidžia autorinių teisių.

2.1.2.2. Papildomos sistemos funkcijos

Papildoma funkcija „Whitelist“ (baltojo sąrašo) :

Autorinėmis teisėmis apsaugotas turinys gali būti naudojamas ir pagal paskirtį. Sudarius sutartį, arba neformalų susitarimą dėl turinio naudojimo su kito domeno savininku, yra poreikis leisti naudotis turiniu ir tretiesiems asmenims. Baltojo sąrašo funkcija leis autoriui nurodyti, kokie domenai turi sutikimą naudotis turiniu ir šiems domenams netaikys įprastų apribojimų.

Papildoma funkcija „Leakproof“ (apsauga nuo nutekėjimo) :

Dažnai autorinėmis teisėmis apsaugotas turinys turi ypatingą svarbą įmonei ir patekęs į nepatikimas rankas gali padaryti daug finansinės žalos. Kaip pavyzdį galime įsivaizduoti įmonę, turinčią slaptus brėžinius, kurie neturėtų patekti už įmonės ribų. Tokiu atveju yra poreikis sekti, šio turinio kelią bei tiksliai nustatyti kas nutekino informaciją, jei toks įvykis vis dėl to atsitiktų. Įskiepio apsaugos nuo nutekėjimo funkcijas leis nurodyti kam buvo siųstas autorinis failas ir atradus saugumo pažeidimą, tiksliai nustatyti pažeidėją.

2.1.3. Duomenų slėpimo algoritmas

Pagal iškeltus bendruosius reikalavimus matome, jog bus reikalingas duomenų slėpimo algoritmas. Analizės dalyje buvo apžvelgti populiariausi duomenų slėpimo bei šifravimo būdai, jų taikymo sritys ir specifikos, tačiau vienu būdu šis sprendimas gali neapsiriboti, dėl skirtingų kriterijų, kuriuos turi atitikti algoritmai naudojami skirtinguose scenarijuose. Toliau pateikiami kriterijai duomenų slėpimo algoritmams :

Scenarijus 1 :

Norime paslėpti paveikslėlyje autoriaus informaciją, tikėdamiesi atpažinti šį paveikslėlį ir pranešti autoriui apie pažeidimą.

Kriterijai :

- Algoritmo tvirtumas – algoritmas turi būti atsparus failo modifikacijoms, tokioms kaip ištampymas, karpymas, suspaudimas ar kitas apdirbimas, nes norime kad autoriaus informacija būtų atpažįstama net ir failą modifikavus.

Scenarijus 2 :

Norime paslėpti paveikslėlį kitame paveikslėlyje įterpiant raktą pradinio paveikslėlio atstatymui, tikėdamiesi apsaugoti dokumentą nuo neleistino jo naudojimo.

Kriterijai :

- Algoritmas būti lengvai jungiamas su įterpiamu raktu informacijai užšifruoti – pagrindinis šio scenarijaus kriterijus, nes turinio apsaugojimas raktu yra būtinas norint patikimai apsaugoti paslėptą turinį nuo jo atskleidimo.

- Talpumas – algoritmas turi leisti paslėpti kuo didesnį informacijos kiekį, nes paveikslėlyje bus slepiama visa jo kopija, ji turi išlikti kuo geresnės kokybės, todėl slepiamų duomenų kiekis yra ypač svarbus šiam scenarijui.

Pagal analizės dalyje apžvelgtus algoritmų pasirinkimus buvo rastas tinkamas algoritmas norimoms autorinių teisių apsaugos priemonėms įgyvendinti.

Scenarijus 1 – algoritmas „F5“.

Šis algoritmas yra pakankamai pažangus ir turi šiam scenarijui reikalingų ypatybių, tokių kaip steganografinės informacijos sklidimo sumažinimas, padidintas atsparumas dydžio keitimui ar pasukimui bei kitoms modifikacijoms.

Scenarijus 2 – algoritmas „F5“.

Kaip minėta praeitame scenarijuje, algoritmas yra pakankamai pažangus, todėl taip pat turi ir duomenų šifravimo raktu galimybę, tai yra būtina antrajam scenarijui.

Tačiau,

šis algoritmas turi nedidelę įterpiamų duomenų talpos galimybę. Taip yra todėl, jog jame yra naudojamos standartinės JPEG kvantavimo lentelės, kurios yra optimizuotos paveikslėlio kokybei palaikyti, pagal standartinį JPEG kompresijos panaudojimą, o ne užtikrinti didelį įterpiamos slaptos informacijos kiekį, ko yra siekiama steganografinių metodų atveju.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

17 pav. Standartinė JPEG kompresijos algoritme naudojama kvantavimo lentelė

Remiantis atliktu JPEG kvantavimo lentelių tyrimu [5] DCT paremtų steganografinių metodų talpai didinti galime pasitelkti optimizuotas kvantavimo lenteles, tokiu būdu po diskrečiosios kosinuso transformacijos sukuriant daugiau vietos slaptai informacijai įterpti. Tą galime atlikti todėl, jog po DCT didelė dalis koeficientų lieka su 0 reikšmėmis, o „F5“ algoritmas 0 reikšmių nenaudoja slaptos informacijos kodavimui, todėl praplėtus ne nulinių reikšmių kiekį, naudojant modifikuotą

[5] C.C. Chang, T.S. Chen, L.Z. Chung, A steganographic method based upon JPEG and quantization table modification. Inf. Sci. 141, 123–138 (2002).

kvantavimo lentelę galima padidinti steganografinio algoritmo talpą. 18 pav. pateikiama minėto tyrimo metu naudota modifikuota kvantavimo lentelė :

16	11	10	16	1	1	1	1
12	12	14	1	1	1	1	55
14	13	1	1	1	1	69	56
14	1	1	1	1	87	80	62
1	1	1	1	68	109	103	77
1	1	1	64	81	104	113	92
1	1	78	87	103	121	120	101
1	92	95	98	112	100	103	99

18 pav. C. Chango atlikto tyrimo metu pasiūlyta modifikuota kvantavimo lentelė sėkmingai padidinusi steganografinio metodo talpą

Remiantis optimalių kvantavimo lentelių sudarymo DCT paveikslėlių kompresijai tyrimu [6] optimalią kvantavimo lentelę norimam kompresijos santykiui (CR) galima sudaryti apskaičiuojant A D ir F koeficientų reikšmes, pagal formulę 8x8 dydžio koeficientų lentelėms :

$$\begin{cases} A = 5.43 + 2.15C_R \\ D = 0.0969 - 0.0565C_R + 0.00749C_R^2 \\ F = 1.83 \end{cases}$$

Tuomet gautas koeficientų A D F reikšmes reiktų įstatyti kiekvienam lentelės koeficientui Q(x,y), pagal formulę :

$$Q_{x,y} = A + Dz^F$$

Formulėje x, y – kvantavimo lentelės koeficientų koordinatės, z = x + y Manheteno atstumas, nuo koordinatės (0,0).

Taip pat kvantavimo lentelę galima sugeneruoti eksperimentiniu būdu bandant sudėlioti vienetinius koeficientus taip, jog po steganografinio metodo panaudojimo kompresijos santykis būtų panašus į tą, kuris yra naudojamas standartiniame JPEG kompresijos algoritme.

[6] D.M. Monro, B.G. Sherlock, Optimal quantization strategy for DCT image compression. IEE Proc., Vis. Image Signal Process. 143(1), 10–14 (1996)

2.2. Siūlomas metodas – „F5A“

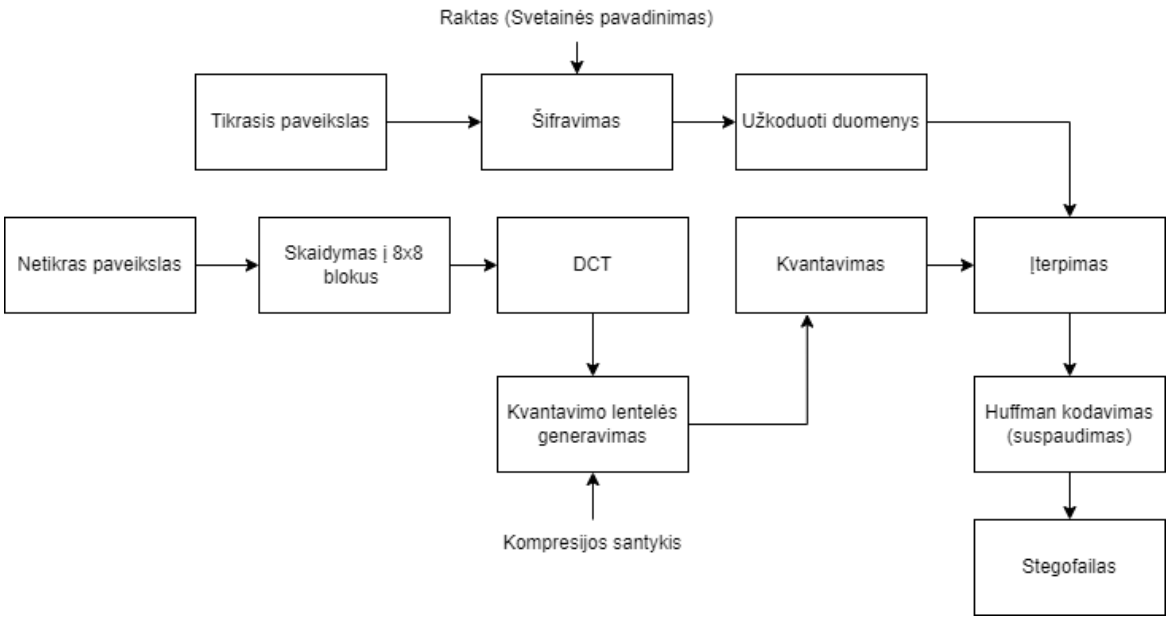
Atsižvelgiant į „F5“ algoritmo talpos apribojimus, siūlomas naujas sprendimas, naudojantis optimalių kvantavimo lentelių generavimą pagal pasirenkamą kompresijos santykį, siekiant padidinti slepiamos informacijos talpą.

Pateikiami algoritmo žingsniai :

1. Nurodomi algoritmui reikalingi duomenys :
    a. Norimas kompresijos santykis (pusiau slapta dedamoji).
    b. Įkoduojami duomenys (Tikrasis paveikslėlis).
    c. Kodavimo raktas – svetainės, kur bus talpinamas paveikslėlis pavadinimas.
2. Nuskaitomi nešyklės duomenys (RGB pikseliai) (nuskaitomas paveikslėlis).
3. Nuskaityti duomenys padalinami į nepersidengiančius blokus po 8x8. Ir naudojama diskrečioji kosinuso transformacija (DCT), suskaidanti blokus į koeficientų matricas.
4. Pagal nurodytą kompresijos santykį generuojama kvantavimo koeficientų lentelė.
5. Pagal sugeneruotą kvantavimo koeficientų lentelę atliekamas kvantavimas.
6. Apskaičiuojama galima talpa, ji palyginama su slaptų duomenų (tikrojo paveikslėlio) užimama talpa.
7. Tikrojo paveikslėlio duomenys užkoduojami raktu.
8. Sugeneruojamas kriptografiškai stiprus atsitiktinių skaičių generatorius iš duoto rakto.

9. Vykdoma permutacija (paskirstomos vietos į kurias bus įrašoma slapta informacija, pagal atsitiktinių skaičių generatorių).
10. Į paskirtas vietas įrašoma slapta informacija (užkoduotas tikrasis paveikslėlis).
11. Vykdomas Hofmano kodavimas, paveikslėlis suspaudžiamas.
12. Gaunamas JPG formato stegofailas.

Grafiškai pateikiami algoritmo žingsniai :



19 pav. Naujo steganografinio algoritmo "F5A" užkodavimo schema

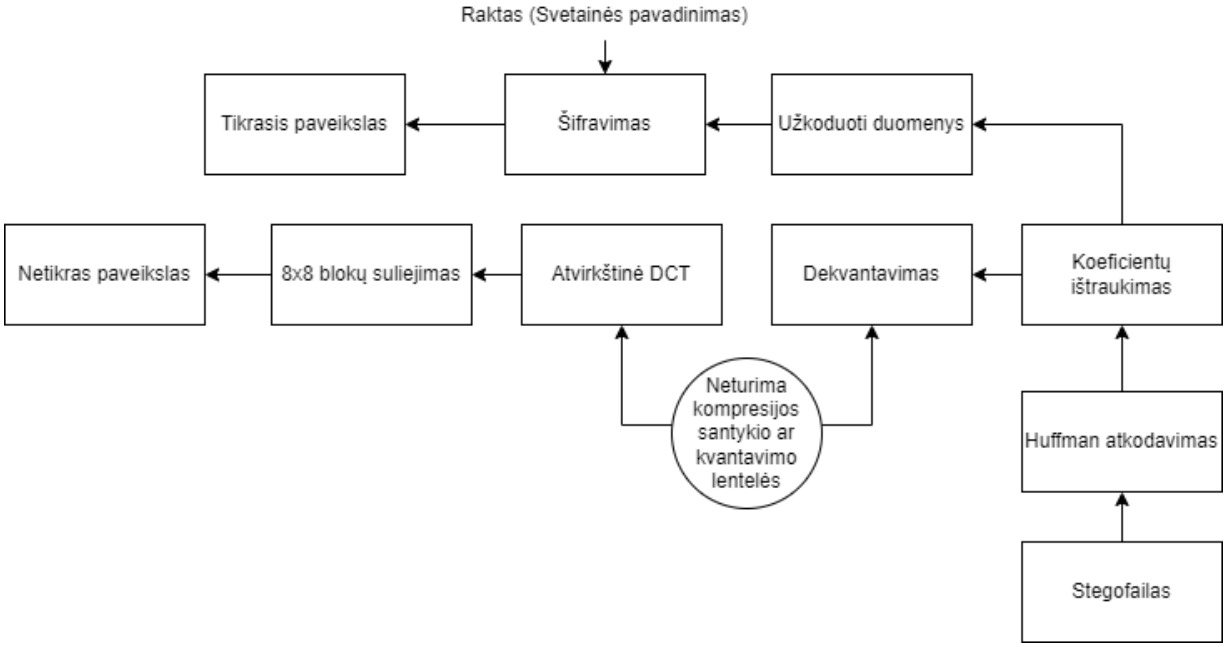
Atkodavimo algoritmas taip pat skiriasi nuo įprasto „F5“ veikimo, pateikiami algoritmo atkodavimo žingsniai :

Pateikiami atkodavimo algoritmo žingsniai :

1. Nurodomi algoritmui reikalingi duomenys :
    a. Gaunamas JPG formato stegofailas.
    b. Kodavimo raktas – svetainės, kur bus talpinamas paveikslėlis pavadinimas.
2. Vykdomas Huffman atkodavimas.
3. Pagal nurodytą raktą randamos vietos kur įterpta slapta informacija, ji atkoduojama naudojant turimą raktą.
4. Gaunamas tikrasis paveikslėlis.

5. Tolimesni žingsniai gali būti atliekami siekiant išgauti neapdirbtą nešyklės paveikslą : dekvantavimas, atvirkščia DCT funkcija, 8x8 blokų suliejimas. Tačiau šį žingsnį, mums yra pravartu apsunkinti, nes būtent atkūrus tikslų nešyklės paveikslą sumažėtų mūsų metodo tinkamumas autorinių teisių apsaugai. Tačiau šį žingsnį jau apsunkina, tai jog norint atkoduoti nešyklės paveikslą tinkamai, mums reikia žinoti koks buvo naudojamas kompresijos santykis, todėl ši dedamoji turi likti dalinai paslaptyje.

Grafiškai pateikiami atkodavimo algoritmo žingsniai :



20 pav. Naujo steganografinio metodo "F5A" atkodavimo schema

2.3. „F5A“ steganografinio metodo panaudojimas autorinių teisių apsaugai

Analizės metu buvo išgryninti reikalavimai, jog naujas autorinių teisių apsaugos metodas turėtų ne tik pranešti apie autorinių teisių pažeidimą, tačiau ir apsaugoti turinį nuo neteisėto jo panaudojimo. Šį reikalavimą išpildyti yra sunku, dėl keleto priežasčių. Visų pirma užkodavus paveikslėlį ir jį išplatinus tokiu būdu, būtų paveikslėlis būtų apsaugomas iki pirmojo jo atkodavimo, kadangi atkodavus paveikslą gauta informacija yra niekaip neapsaugota nuo tolimesnio jos išnaudojimo. O jei pasirenkame neslėpti paveikslėlio, o tik įterpti autoriaus informaciją jame, tuomet galime atpažinti neteisėtą naudojimą, tačiau neteisėtos veiklos nesustabdome. Siūlomas autorinių teisių apsaugos metodas, remiasi naršyklės įskiepiu, kuris galėtų aptikti pavogtą turinį ir apsaugoti jį nuo neteisėto naudojimo.

Metodas sudarytas iš dviejų dalių :

1 – Paveikslėlio užkodavimo. (Atlieka autorius).

2 – Paveikslėlio atkodavimo ir tikrinimo. (Atlieka naršyklės įskiepis).

Paveikslėlis yra užkoduojamas taip, jog iš jo nebūtų galima išgauti tikrojo paveikslėlio, tačiau vartotojui vizualiai to nesimatytų.

Naudojami sutartiniai ženklai :

F – visiems žinomi duomenys, skirti patvirtinimui, jog naudojamas steganografinis algoritmas.

O – tikrasis paveikslėlis, kurį norima apsaugoti nuo neteisėto panaudojimo.

C – nešyklė, bet koks nesvarbus paveikslėlis.

O(F) – į tikrąjį paveikslėlį įterptas patvirtinimo duomenų failas.

C(O(F)) – į nešyklę įterptas tikrasis paveikslėlis, kuriame užkoduotas patvirtinimo duomenų failas.

A – įprastas, niekuo neapdorotas paveikslėlis.

Paveikslėlio užkodavimo eiga.

1. Naudojant naują „F5A“ algoritmą steganografiniu būdu į originalų paveikslėlį O įterpiamas patvirtinimo duomenų failas. Įterpiant yra naudojamas raktas (verifyKEY), kuris yra visiems žinomas. Gaunamas O(F).

2. Tuomet į nešyklės failą C tokiu pat būdu naudojant „F5A“ įterpiamas pirmame žingsnyje gautas O(F), šį kartą naudojamas raktas yra svetainės pavadinimas (domain), kurį galės patvirtinti mūsų įskiepis. Į C įterpus O(F) gauname C(O(F)).

Paveikslėlio atkodavimo eiga.

1. Atkodavime turime patikrinti, ar paveikslėlis yra įprastas, ar apdirbtas „F5A“ algoritmu, taip pat nustatyti ar jis gali būti rodomas svetainėje.

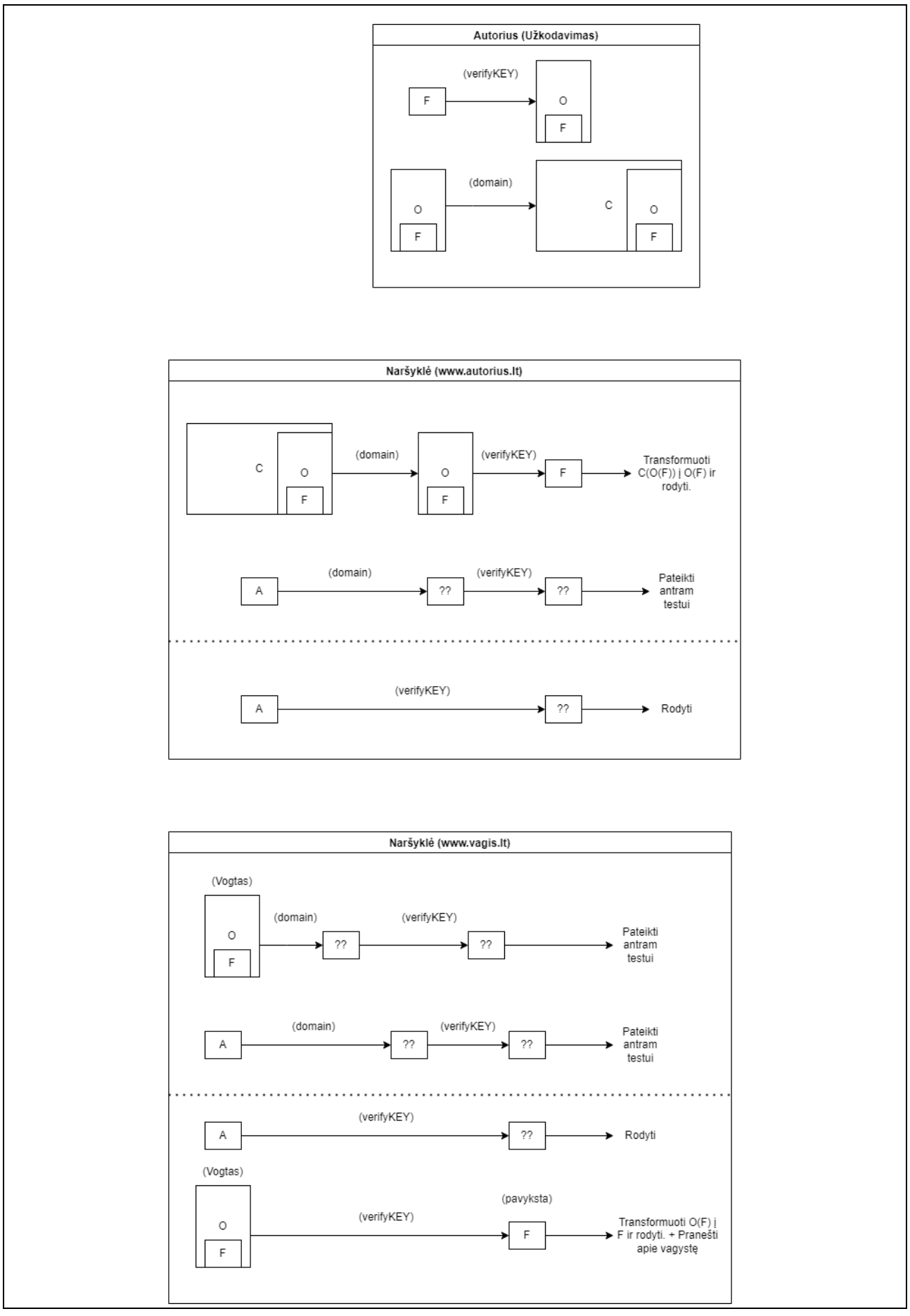
2. Pirmiausia tikriname, ar paveikslėlis yra užkoduotas pagal C(O(F)) schemą. Tai yra ar jį atkoduoti galime panaudojus pirmam atkodavimui svetainės vardą (domain) ir antram atkodavimui patvirtinimo raktą (verifyKEY), patikriname ar gauname patvirtinimo duomenis F. Jei taip, atkoduojame C(O(F)) į O(F) ir pateikiame vartotojui originalų paveikslėlį su įterptu patvirtinimo duomenų ženklu. Vizualiai skirtumo nesimato nuo originalaus paveikslėlio.

3. Jei esame turinio vagys, bandydami parsisiųsti paveikslėlį, gautume C(O(F)). O jei matome jau atkoduotą paveikslėlį O(F), galime padaryti ekrano nuotrauką ir taip gauti O(F). Tačiau apie šiuos atvejus yra apgalvota ir tam pritaikytos apsaugos priemonės.

4. Tačiau svarbu pagalvoti ir apie įprastus svetainės paveikslėlius, kaip juos atpažinti ir leisti rodyti, uždrausti nelegalų O(F) rodymą.

5. Tam reikalingas antras testas, kuris, nepavykus pirmajam patikrintų, ar failas yra apdirbtas „F5A“ algoritmu. Bandoma atkoduoti su (verifyKEY) ir šiam veiksmui pavykus, galime teigti, jog turinys yra vogtas, tačiau jei atkoduoti su (verifyKEY) nepavyksta, tuomet paveikslėlis yra įprastas ir jo rodymas yra leidžiamas.

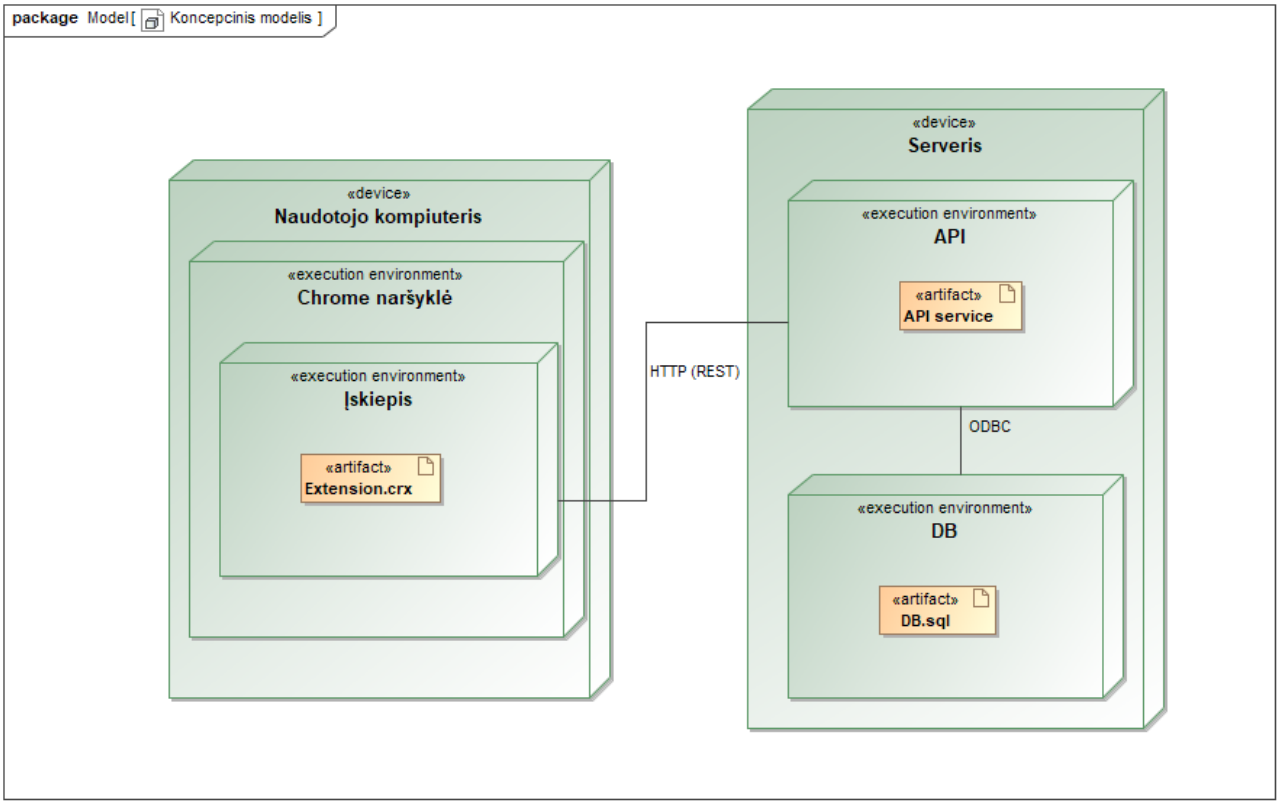
Pateikiama grafinė „F5A“ algoritmo panaudojimo autorinių teisių apsaugai schema :



1 lentelė „F5A“ algoritmo panaudojimo autorinių teisių apsaugai schema

2.4. Koncepcinis modelis

Šiame skyriuje yra pateikiamas siūlomo sprendimo koncepcinis modelis, siekiant parodyti kokie komponentai bus reikalingi sprendimui bei komunikacijai tarp jų :

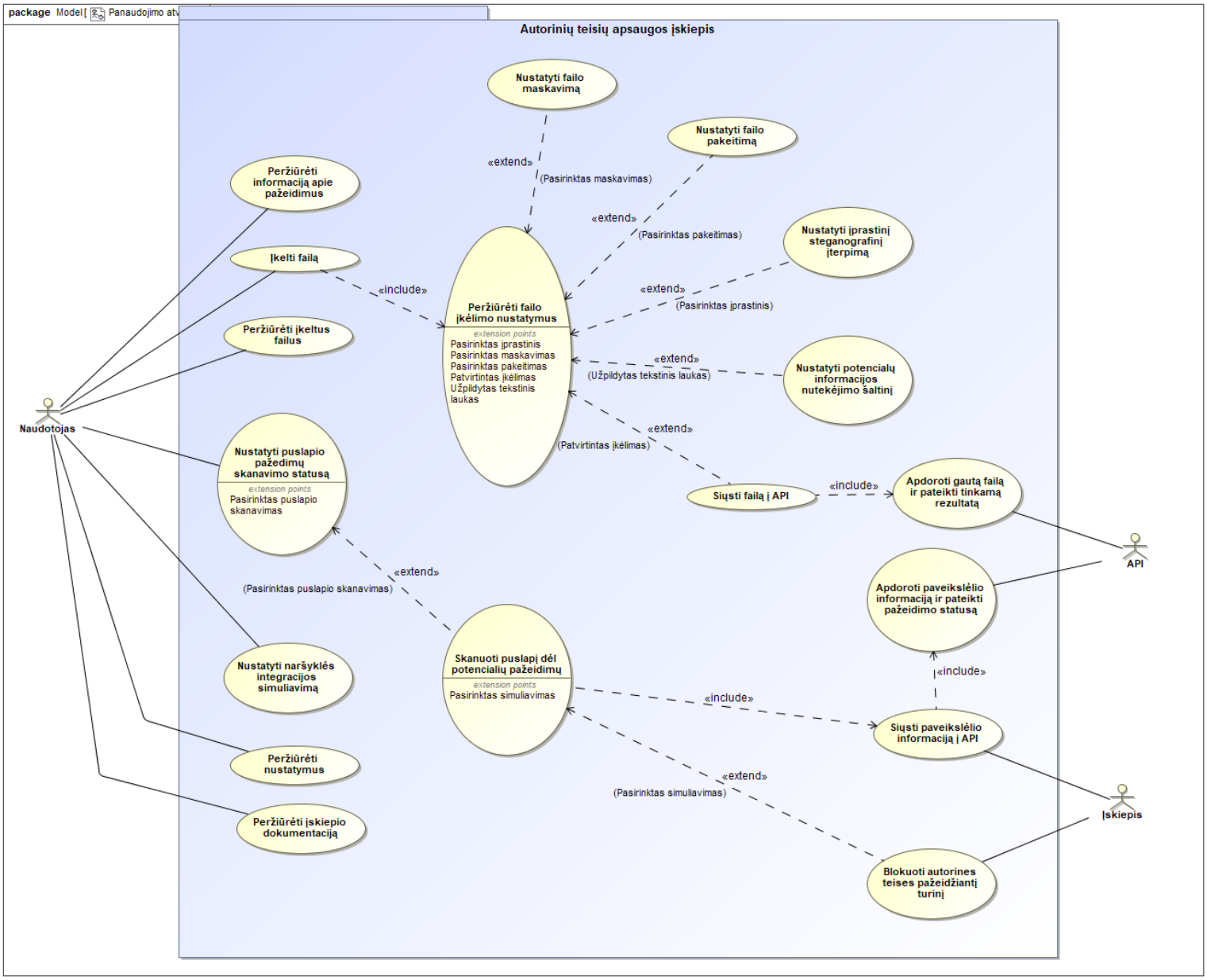


21 pav. Įskiepio diegimo diagrama rodanti skirtingus sprendimo komponentus bei komunikaciją tarp jų

Sprendimo įgyvendinimui bus reikalingi trys pagrindiniai komponentai – Įskiepis, kuris bus naudojamas potencialių autorinių teisių pažeidimų paieškai naršomame puslapyje, Aplikacijos programavimo sąsaja (API), kuri bus reikalinga norint įkelti paveikslėlį ir duomenis apie jo autorių bei pasirinkimus kaip bus saugomas duotas paveikslėlis. API taip pat bus atsakingas už bendravimą su trečiuoju komponentu – duomenų baze (DB), kuri bus reikalinga saugoti duomenims apie paveikslėlio autorius ir įkeltus paveikslėlius.

2.5. Panaudos atvejai

Šiame skyriuje yra pateikiami siūlomo sprendimo panaudos atvejai, siekiant parodyti kokius veiksmus gali atlikti vartotojas naudojantis sukurtu įskiepiu :



22 pav. Įskiepio panaudojimo atvejų diagrama

Įskiepio pagrindinis funkcionalumas yra skanuoti puslapį ir apdoroti informaciją apie rastus paveikslėlius kurie pažeidžia autorines teises. Ši funkcija yra vykdoma automatiškai, kiekvieną kart atsidarius naują puslapį, jei naudotojas nėra išjungęs šio funkcionalumo. Atsidaręs pagrindinį įskiepio langą vartotojas galės peržiūrėti informaciją apie rastus pažeidimus, matyti dabartiniame puslapyje esančių pažeidimų kiekį bei per visą naudojimo laikotarpį rastų autorines teises pažeidžiančių paveikslėlių kiekį. Kitas pasirinkimas kurį naudotojas matys tame pačiame lange yra įkelti failą. Pasirinkus šią funkciją bus atveriamas naujas langas su failo įkėlimo nustatymais. Jame pagal naudotojo pasirinkimus bus nustatoma informacija prieš įkeliant failą. Pažymėjus reikiamas savybes ir paspaudus siųsti failas bus siunčiamas į API.

API turės atlikti dvi pagrindines funkcijas : gavus naują failą, užregistruoti jo informaciją autorinių teisių apsaugos duomenų bazėje arba gavus informaciją apie puslapyje rastą paveikslėlį, pateikti informaciją ar šis paveikslėlis pažeidžia autorines teises. Šiame žingsnyje gali iškilti problemų dėl

per dažno kreipimosi į API, nes kiekvienas puslapis gali turėti daug paveikslėlių ir kaskart reikės kreiptis į API, tai bus bandoma išspręsti kuriant atskirus servisus failo įkėlimui bei patikrinimui, kadangi failo patikrinimui skirta dalis galės būti kešuojama taip didinant sprendimo spartą.

Paskutinis komponentas yra duomenų bazė. Ji bus skirta duomenims apie autorius bei jų įkeltus paveikslėlius saugoti.

2.6. Grafinės vartotojo sąsajos prototipas

Autorinių teisių apsaugos įskiepio projektas bus sudarytas iš dviejų dalių : API ir naršyklės įskiepio. API grafinės vartotojo sąsajos, kaip žinome neturės, tačiau naršyklės įskiepio grafinė vartotojo sąsaja bus kuriama.

Įskiepis bus sudarytas iš šių pagrindinių langų :

- Įskiepio lango (santraukos lango, rodančio pagrindinę puslapio statistiką ir keletą pagrindinių funkcijų)
- Failo įkėlimo bei parinkčių lango (santraukos lange bei pilno puslapio dydžio)
- Nustatymų lango (pilno puslapio dydžio lango, su visomis likusiomis sprendimo funkcijomis bei pateikiama pilna statistika apie rastus pažeidimus)
- Pagalbos lango (pilno puslapio dydžio lango su įrankio dokumentacija, naudojimosi instrukcijomis)

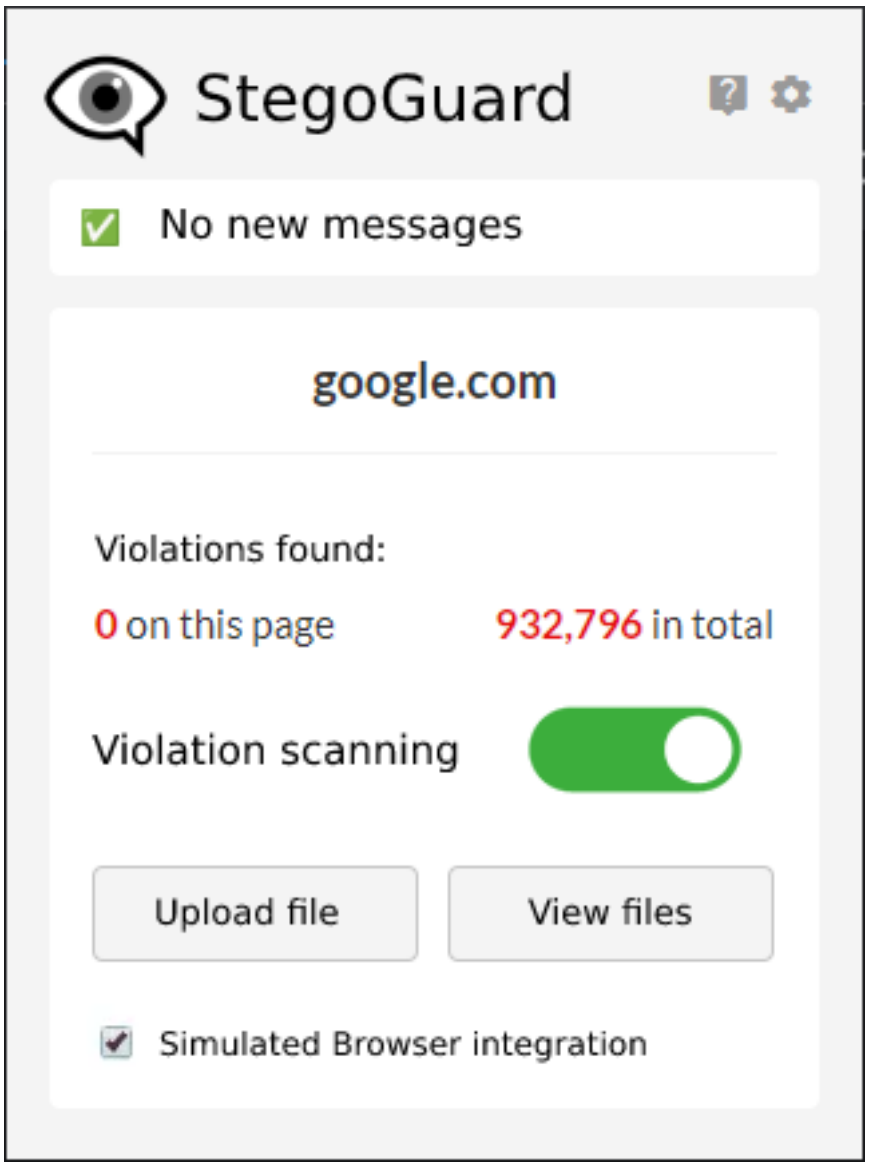
Įskiepio lango viršuje bus rodomas kuriamo sprendimo logotipas ir pavadinimas (bus tikslinama).

Sprendimo pavadinimas „**StegoGuard**“ ir pirminis logotipas (emoji) - 👁‍🗨.

Taip pat lango viršuje bus dvi ikonos su nuorodomis į pagalbos bei nustatymų langus.

Toliau bus įskiepio pranešimų laukas, jame bus vaizduojami klaidos arba pasisekimo pranešimai.

Po pranešimų lauko turėsime funkcijų lauką, jame bus atvaizduojamas skanuojamos (dabartinės) svetainės adresas bei aptiktas pažeidimų kiekis. Taip pat šiame lauke bus galima nustatyti pagrindinius įskiepio nustatymus. Matysime puslapio skanavimo įjungimo / išjungimo mygtuką, nuorodą į mūsų įkeltų failų sąrašą, nuorodą į failo įkėlimo parinkčių langą. Taip pat turėsime galimybę pažymėti, jog bus naudojamas „Simulated Browser integration“, integracijos į naršyklę režimas.



23 pav. Įskiepio lango grafinės vartotojo sąsajos prototipas

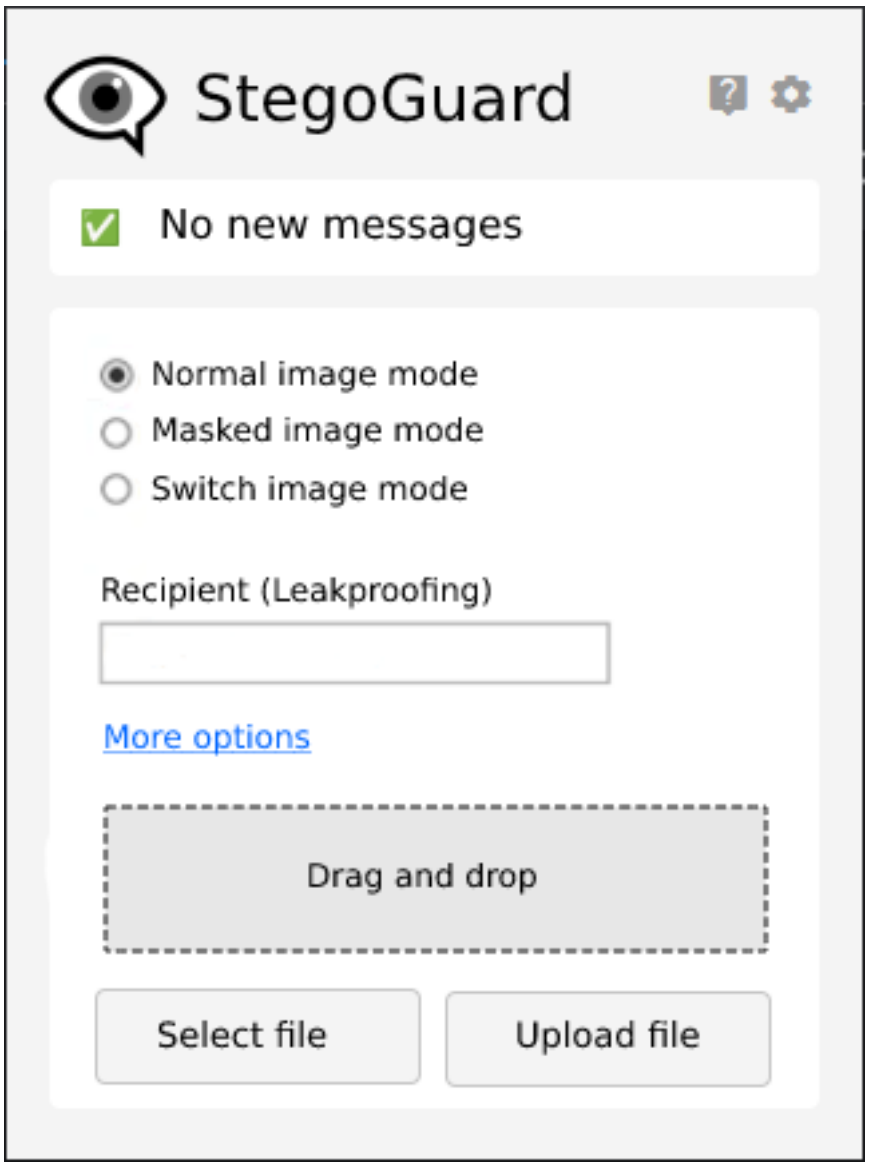
Paspaudus „Upload file“ bus matomas failo įkėlimo bei parinkčių langas.

Šiame lange bus galima pasirinkti failo apsaugos būdą, pasirenkant tarp normalaus (Normal), kuris paprasčiausiai įterps autoriaus informaciją į paveikslėlį, maskavimo (Masked), kuris gražins užmaskuotą paveikslėlį kartu su atstatymo raktu ir pakeitimo (Switch) būdo, kuris gražins vieną paveikslėlį, užmaskuotą kito pavidalu, kartu su jo atstatymo raktu.

Taip pat šiame lange galėsime nurodyti kam bus siunčiamas paveikslėlis, siekiant apsaugoti informaciją nuo nutekėjimo. (Leakproofing laukas).

Jei norėsime naudoti kitas įkėlimo funkcijas būsime nukelti į nustatymų lange esantį „Advanced Upload“ puslapį.

Galiausiai lange numatomai bus „drag and drop“ funkcionalumas, siekiant suteikti patogų failo įkėlimą.



24 pav. Įskiepio failų įkėlimo lango grafinės vartotojo sąsajos prototipas

Nustatymų lange bus pateikiamos visos esamos įskiepio funkcijos, įskiepio statistika apie autorinių teisių pažeidimus, pagalbos puslapis bei išsamus failų įkėlimo puslapis.

Statistikos dalyje planuojama pateikti rastų autorinių teisių pažeidimo kiekį, tiek kitų autorių kūriniams tiek mūsų įkeltiems failams, taip pat pateikiami mėnesio / savaitės atrastų pažeidimų grafikai.

Išsamaus failų įkėlimo puslapyje pateikiamos tokios pačios funkcijos kaip ir santraukos failų įkėlimo dalyje, tačiau su papildomomis funkcijomis : Galimybe nurodyti nuo pažeidimų atleidžiamų puslapių adresus (Whitelist), pranešimo apie pažeidimą el. paštu nustatymai bei kitos parinktys.

Taip pat viena iš puslapių matysime ir naršyklės integracijos simuliacijos parinktis. Bus leidžiama pasirinkti autorines teises pažeidžiančių kūrinių slėpimo būdą : Default (numatytąjį), Masked (maskuojamą), arba Blocked (blokuojamą).

Pagalbos puslapyje bus pateikiama visa reikalinga dokumentacija sėkmingam įskiepio naudojimui, dažniausiai užduodami klausimai bei pagalbos kontaktai.

Išvados ir uždaviniai

1. Pasinaudojant analizės dalyje gautais rezultatais buvo nustatytas poreikis naujam autorinių teisių apsaugos sprendimui bei išsikelti bendrieji reikalavimai :

   - Atpažinti neteisėtai naudojamą turinį, ir pranešti apie tai autorinių teisių savininkui.
   - Apsaugoti atpažintą turinį nuo neteisėto jo panaudojimo.

2. Išsikeltiems reikalavimams įgyvendinti buvo projektuojamas sprendimas, kurio pagrindinė mintis yra ne tik pranešti autoriui apie jo teisių pažeidimą, tačiau ir aktyviai apsaugoti jo turinį nuo neteisėto panaudojimo, apsibrėžti du būdai kaip tai galėtų būti įgyvendinama :

   - Integracija į naršyklę, turinio blokavimas (simuliacija)
   - Turinio apsaugojimas raktu

3. Nustačius pagrindinius autorinių teisių apsaugos scenarijus buvo iškelti reikalavimai steganografiniam algoritmui, kuris bus naudojamas įskiepyje. Reikalavimams įgyvendinti suprojektuotas naujas steganografinis metodas „F5A“, kuris remiasi „F5“ algoritmu, praplėsdamas jo talpą naudojant generuojamas kvantavimo lenteles bei pridėdamas naujų savybių, tokių kaip apsauga nuo nešyklės turinio atstatymo bei atkodavimo metu vykdomas pranešimas apie autorinių teisių pažeidimą.

4. Parengtas siūlomo sprendimo projektas, apibrėžti įskiepio panaudojimo atvejų scenarijai, veikimo konceptas. Sudaryti potencialiai reikalingų duomenų bei grafinės vartotojo sąsajos prototipai.

3. Projekto realizacija

Šiame skyriuje yra pateikiamas naujo steganografinio metodo „F5A“, skirto skaitmeninės medijos apsaugai detalizuotas aprašymas. Taip pat papildomai pateikiamas ir aprašymas naujo autorinių teisių apsaugai skirto įskiepio „StegoGuard“ koncepcijos, kuriame „F5A“ algoritmas bus naudojamas.

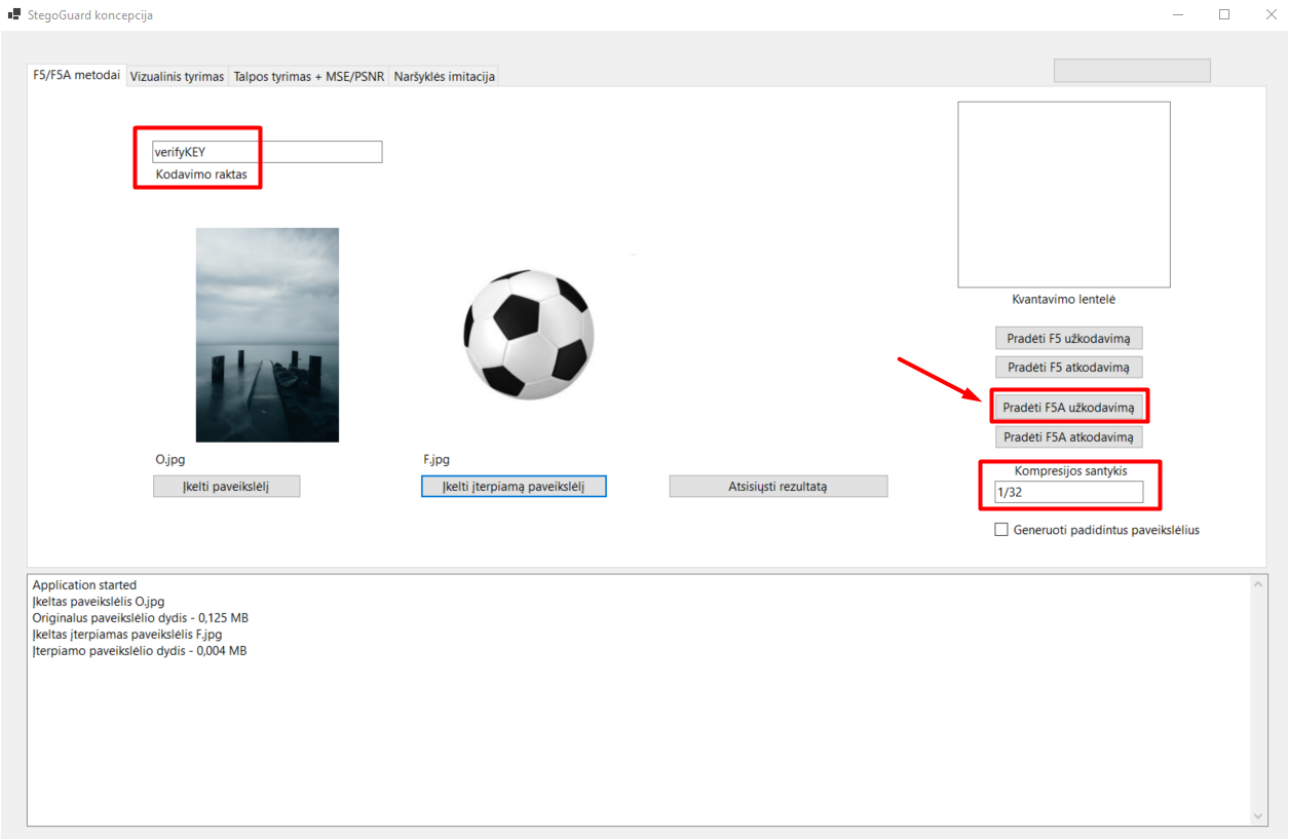
3.1. „F5A“ algoritmo detalizuotas aprašymas

3.1.1. „F5A“ užkodavimas

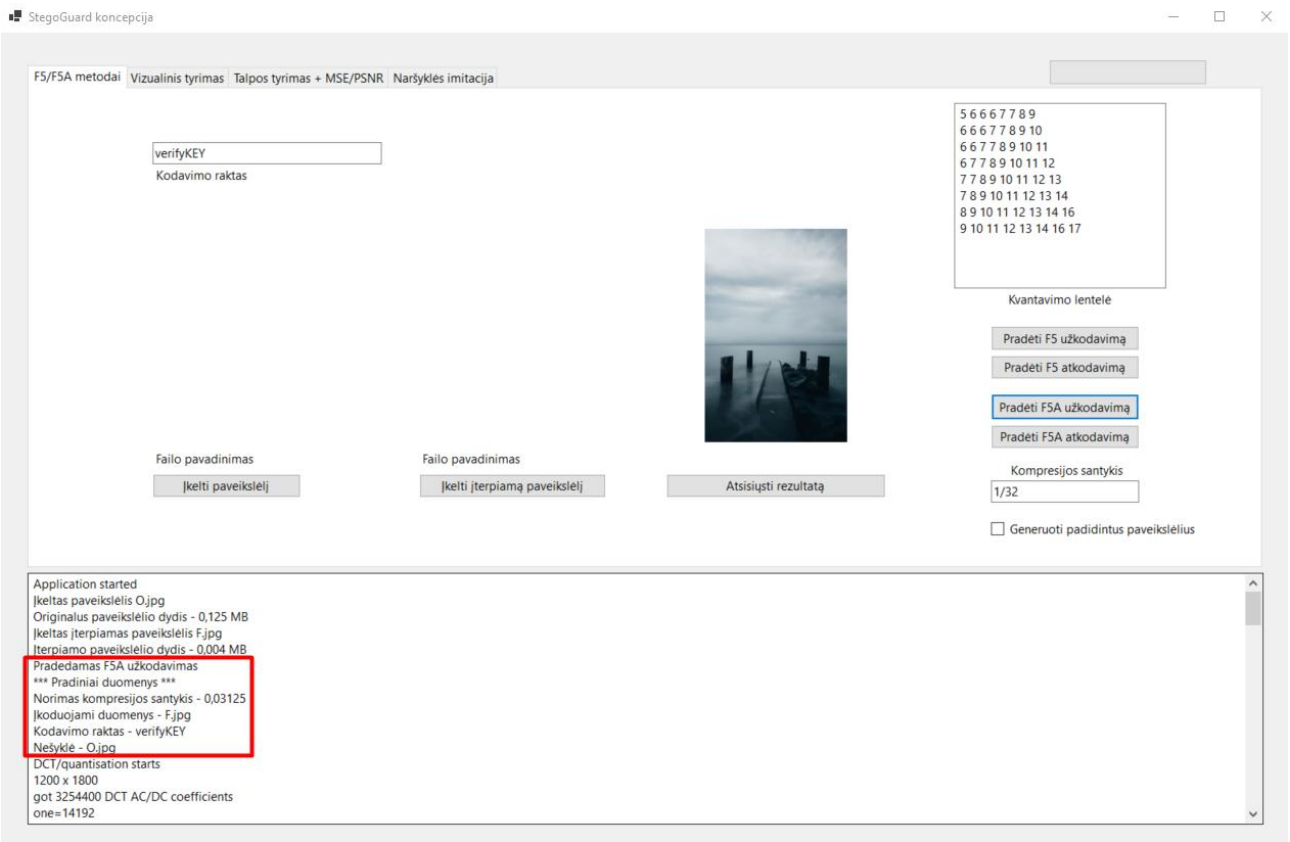
Pateikiamas „F5A“ steganografinio informacijos įterpimo į failą algoritmas

3.1.1.1. Algoritmui reikalingi duomenys

- Norimas kompresijos santykis – skaičius (pusiau slapta dedamoji)
- Įkoduojami duomenys – tikrasis paveikslėlis
- Kodavimo raktas – bet koks žodis ar frazė
- Nešyklė – bet koks paveikslėlis, į kurį bus įkoduojamas tikrasis paveikslėlis



25 pav. StegoGuard įskiepio koncepcijos naudojimas



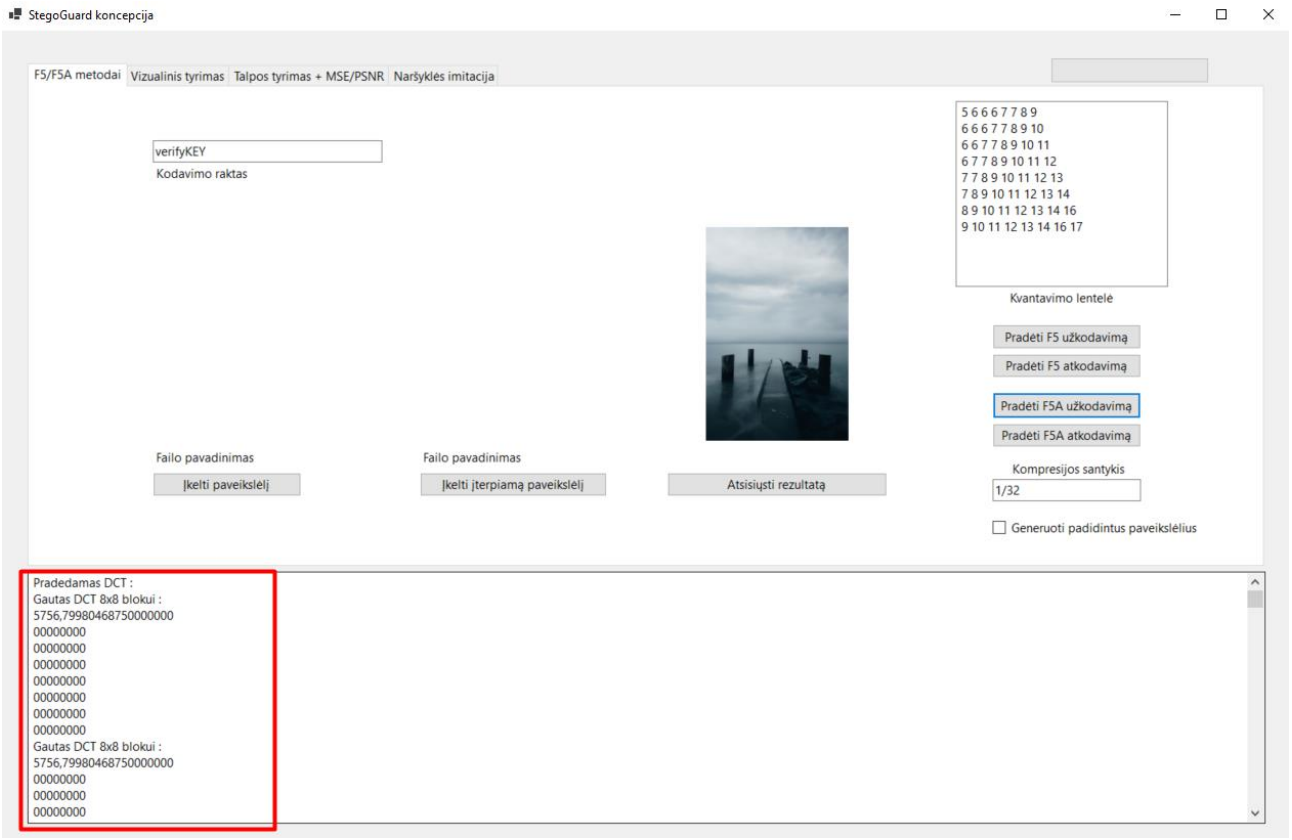
26 pav. Pradiniai duomenys

3.1.1.2. Duomenų nuskaitymas ir transformavimas DCT

Pirmiausia yra nuskaitomi nešyklės duomenys. Kadangi algoritmas bus naudojamas naršyklės įskiepyje, failo nuskaitymas vykdomas C# kalba. Įkelto failo duomenys turi būti nuskaitomi į bitų masyvo struktūrą. Ši struktūra egzistuoja įvairiose kalbose, tačiau C# ji yra prieinama kalboje esančio File I/O API. Bitų masyvo tipo objektą iš nuskaityto failo gauname panaudodami F5 algoritmo implementaciją. Tokiu pat būdu galime nuskaityti ir tikrojo paveikslėlio duomenis.

Tuomet nuskaityti duomenys (pikselių RGB reikšmės) yra padalinami į 8 x 8 dydžio blokus (galime naudoti masyvą, ar kitą pasirinktą struktūrą).

Išskaidžius duomenis į blokus, šie blokai yra transformuojami naudojant DCT – diskrečiąją kosinuso transformaciją. Ji atliekama tokiu pat būdu, kaip ir JPEG kompresijos algoritme bei Andreas Westfeld‘o „F5“ algoritme. Atlikus šią transformaciją, jos rezultatas yra 8 x 8 dydžio blokai su pikselių dažnio koeficientais.



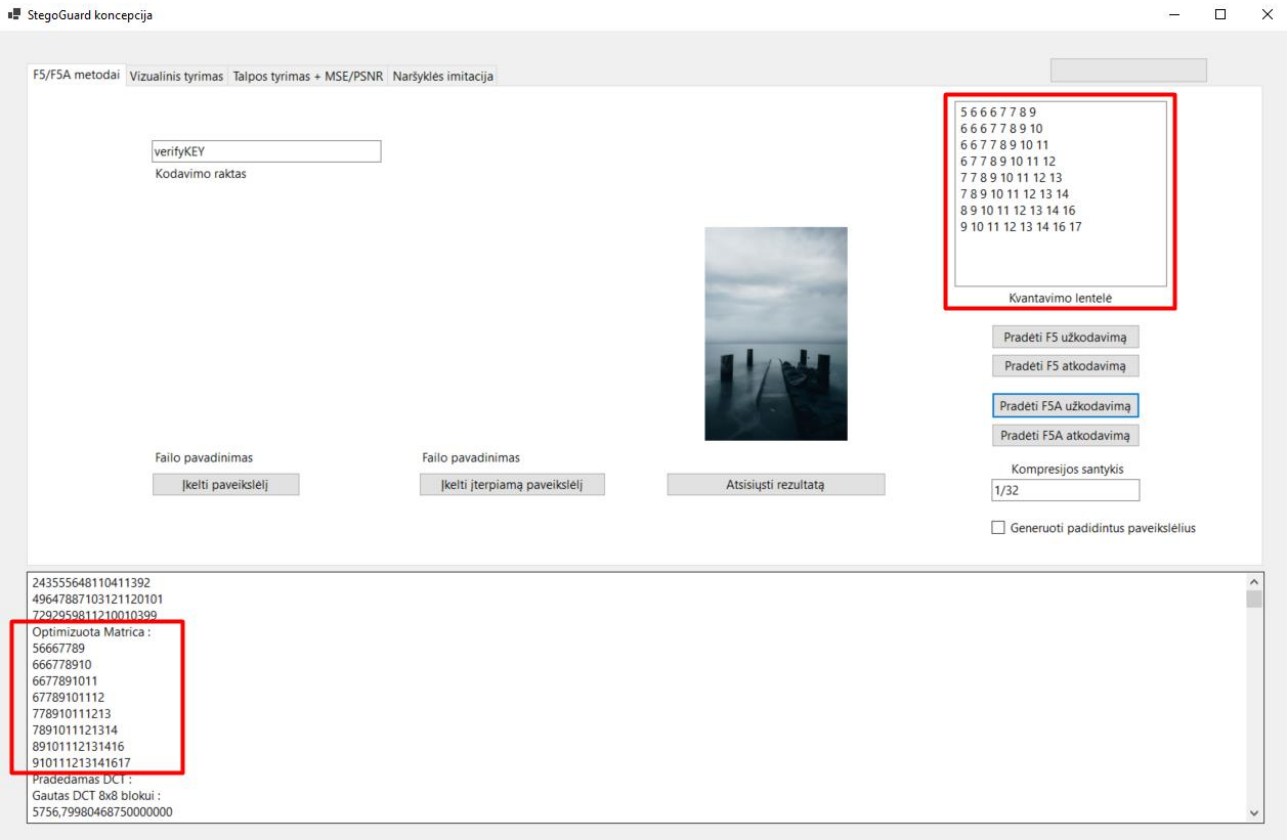
27 pav. DCT transformacijos žingsnis

3.1.1.3. Kvantavimo lentelės generavimas

Išskirtinė „F5A“ metodo savybė – kvantavimo lentelių generavimas, atliekamas norint padidinti „F5“ metodo talpą bei suteikti papildomą savybę metodui – nešyklės išgavimo iš stegofailo apsunkinimą, nežinant kompresijos santykio.

Tuomet pagal projektinėje dalyje nurodytas formules (Skyrius 2.1.3) yra apskaičiuojama kvantavimo lentelė. Šiame žingsnyje yra panaudojamas norimas kompresijos santykis.

Kvantavimo lentelės generavimo žingsnis yra aprašomas programiniame kode, pagal projekte nurodytas formules C# kalba.



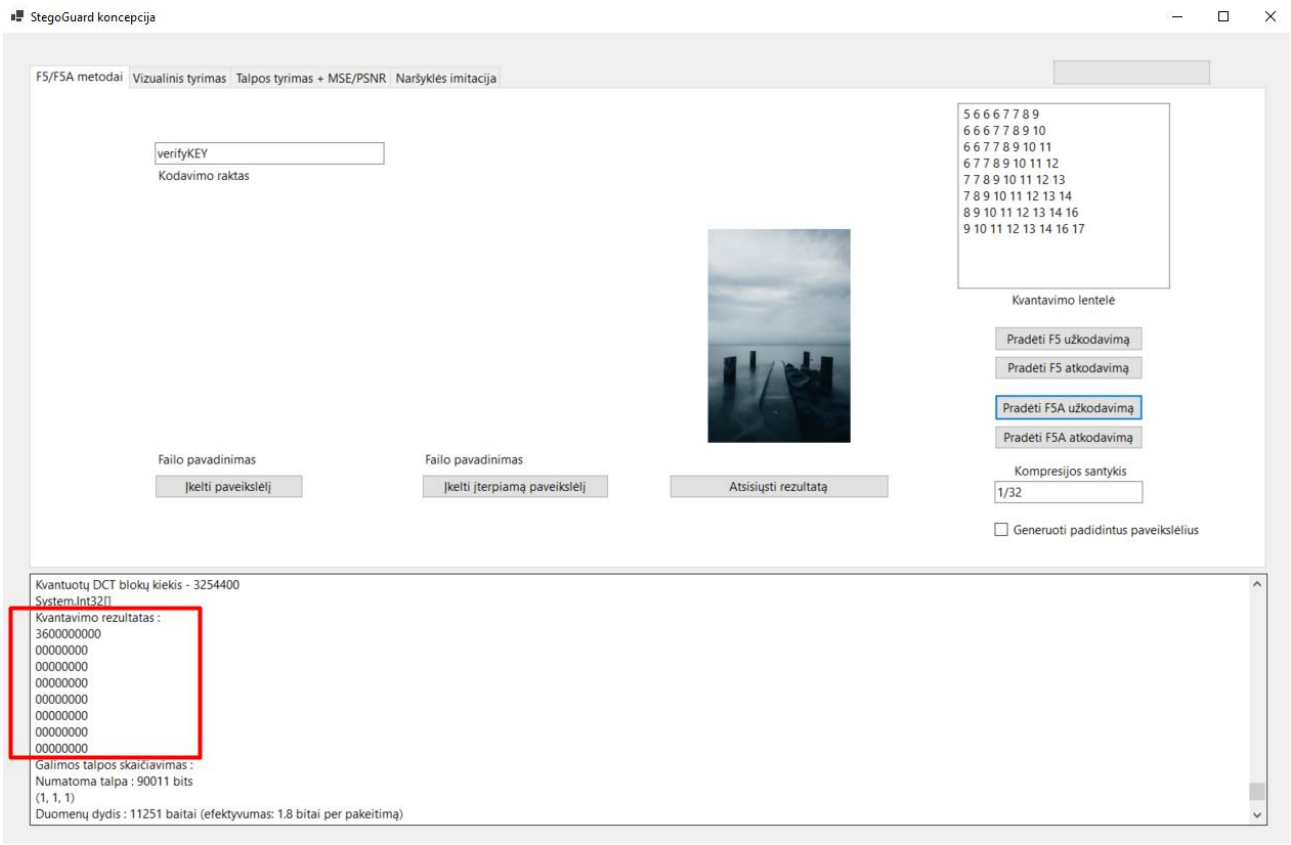
28 pav. Optimizuotos kvantavimo lentelės generavimo žingsnis

3.1.1.4. Kvantavimas

Kvantavimas – kompresijos metodas, kurio metu po DCT gautos koeficientų matricos reikšmės yra dalinamos iš kvantavimo lentelės reikšmių, viena su kita (kiekvienai matricos reikšmei).

Šiam veiksmui atlikti galime naudoti esamą F5 algoritmo implementaciją. Svarbu, jog kvantavimo metu būtų galimybę nurodyti savo kvantavimo lentelę, nes, kvantavimas naudojant standartinę kvantavimo lentelę yra netinkamas „F5A“ algoritmo realizacijai.

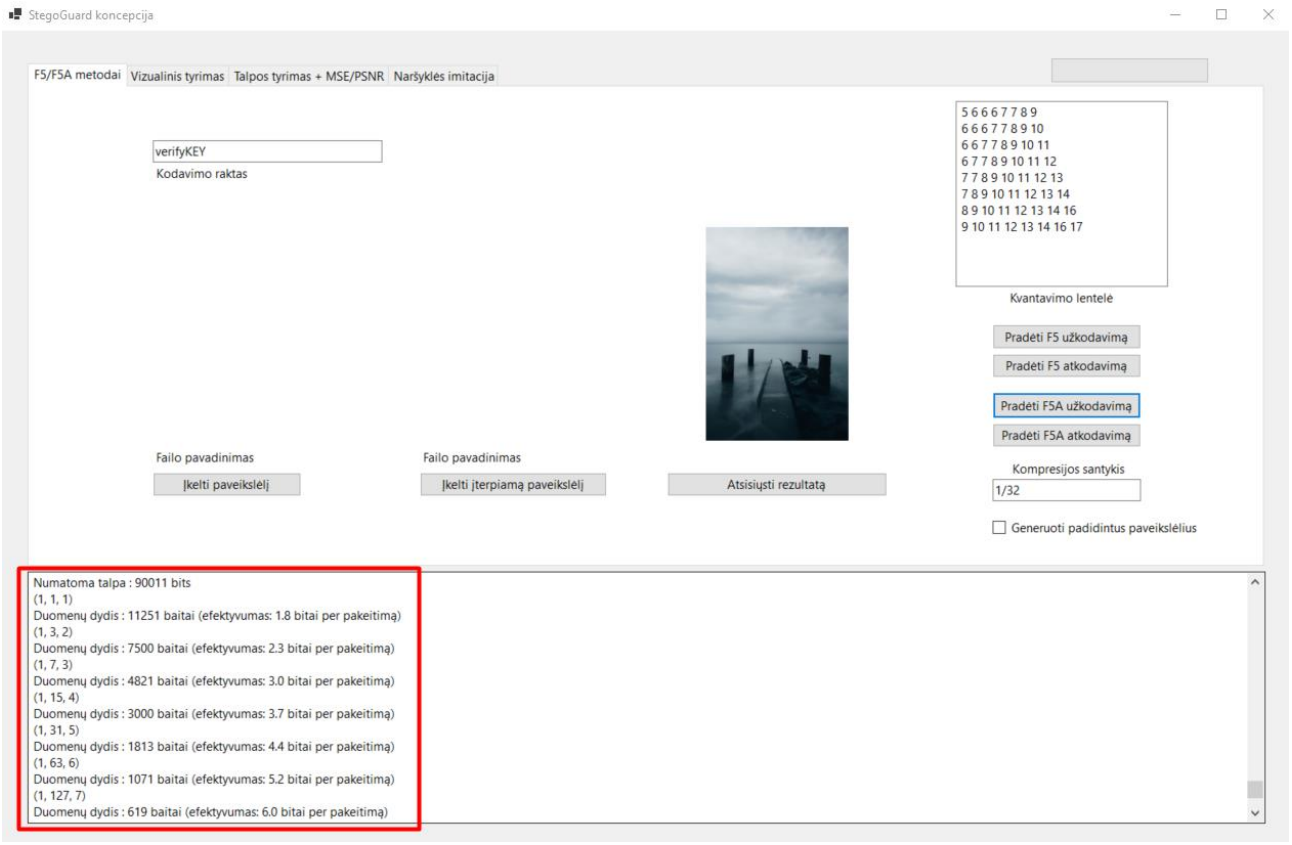
Atlikus kvantavimo žingsnį, yra gaunamos koeficientų matricos, su pagrinde kairiame viršutiniame kampe pasiskirsčiusiomis reikšmėmis, daugelis kitų koeficientų tampa reikšme 0.



29 pav. Kvantavimo žingsnis

3.1.1.5. Galimos talpos apskaičiavimas

Prieš vykdant tolimesnius žingsnius, reikia atlikti galimos talpos apskaičiavimą mūsų kvantuotose matricose. Šio veiksmo esmė yra sužinoti ar galime įterpti mūsų dydžio failą (tikrasis failas) į nešyklę. Veiksmui atlikti C# kalboje galime naudoti esamą „F5“ algoritmo implementaciją, kurioje yra skaičiuojamas baitų kiekis ir palyginamas su kvantuotose matricose galimų koduoti reikšmių kiekiu. Pavadinkime įterpiamos žinutės ilgį (baitų kiekį) simboliu – „**K**“.



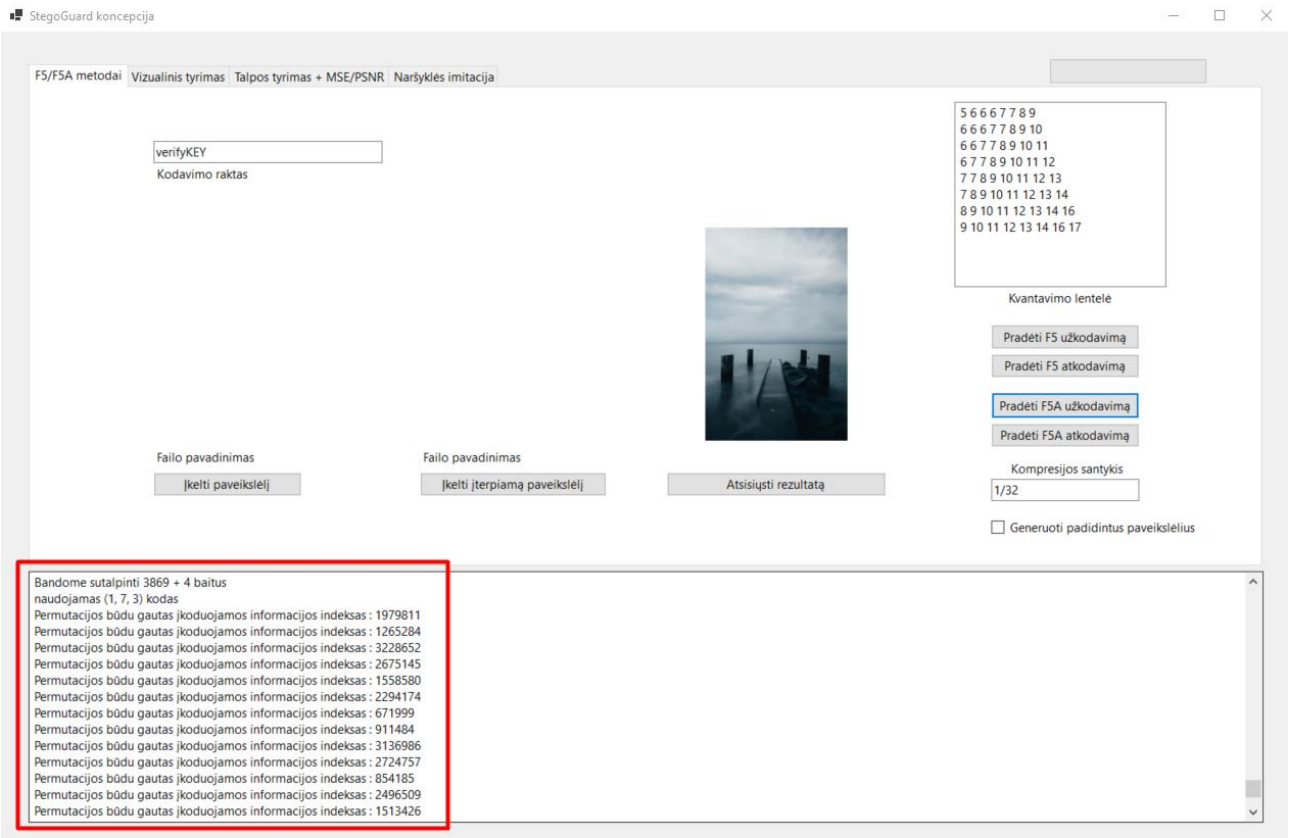
30 pav. Talpos skaičiavimo žingsnis

3.1.1.6. Permutacija

Prieš atliekant permutaciją – užšifruoto paveikslėlio reikšmių paskirstymo per kvantuotą matricą vietų generavimą, sugeneruojamas kriptografiškai stiprus atsitiktinių skaičių generatorius iš kodavimo rakto.

Vykdome permutaciją su sugeneruotais atsitiktiniais skaičiais ir koeficientų kiekiu (0 reikšmės taip pat įtraukiamos).

Permutacijos metu gauname vietas į kurias norėsime įterpti savo užšifruoto paveikslėlio reikšmes.



31 pav. Permutacijos žingsnis

3.1.1.7. Matricos kodavimas

Šiame žingsnyje įterpiame šifruoto paveikslėlio informaciją į mūsų kvantuotas matricas.

Prieš pradedant kodavimą apskaičiuojame buferio BUFF (masyvo kuris dalyvaus kodavime) ilgį :

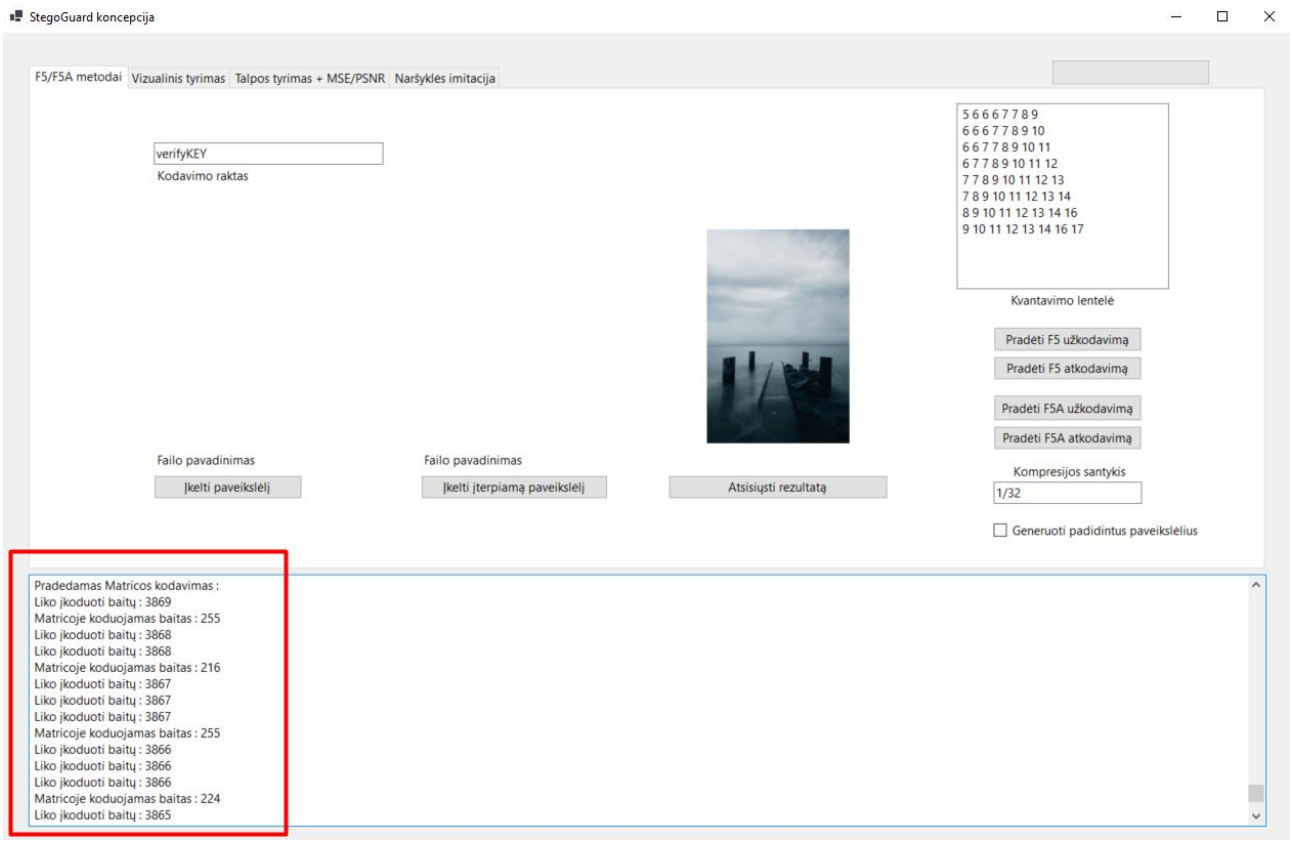
$$N = 2^K - 1$$

Įterpimas vykdomas tokiu būdu :

1. Užpildome BUFF su N kiekiu nenulinių koeficientų iš mūsų koeficientų matricos.
2. Vykdome XOR tipo maišos funkciją minėtą (1.4.2.4 skyriuje) BUFF masyvui. Atlikus maišos funkciją turėtume gauti reikšmę HASH, kurios ilgis K (užimti tiek bitų vietų).
3. Pridedame vieną po kitos K kiekį šifruoto failo reikšmių prie HASH. Gauname reikšmė SUM.
4. Jei sudėjus bitus SUM = 0, BUFF nekeičiame, kitu atveju SUM reikšmes priskiriame BUFF masyvui. Dėliojame SUM[0...pabaiga] į masyvą BUFF[0...N].
5. Patikriname ar BUFF masyve turime nulio reikšmių. Jei taip, 0 pakeičiame tolimesnėmis reikšmėmis iš nenulinių koeficientų. (Kartojame nuo 1 punkto). Jei neturime nulio reikšmių, tuomet imame naują N kiekį nenulinių koeficientų ir kartojame nuo 1 punkto, iki kol nebeliks informacijos kurią reikės įkoduoti.

Matricos kodavimo rezultatas – pilnas reikšmių buferis.

Šiam veiksmui atlikti bus naudojamos esamos „F5“ algoritmo implementacijos.



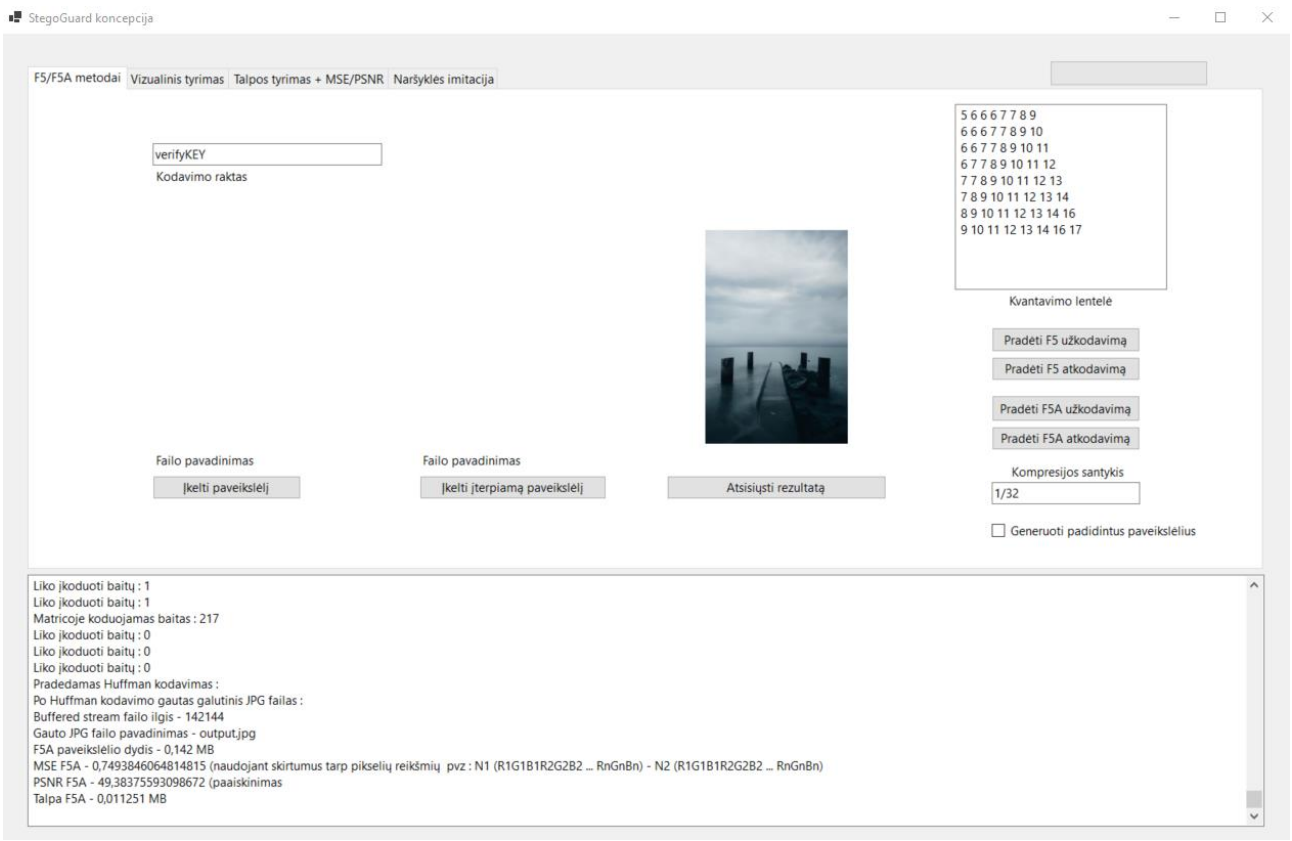
32 pav. Matricos kodavimo žingsnis

3.1.1.8. Hofmano kodavimas

Paskutinis žingsnis yra Hofmano kodavimas. Šis žingsnis leidžia suformuoti mūsų buferį pripildytą matricos reikšmių į binarinį kodą, iš kurio gauname mūsų failo duomenis JPG formatu.

Šis žingsnis bus atliekamas esamomis „F5“ algoritmo implementacijos priemonėmis.

Atlikus Hofmano kodavimą gaunamas galutinis JPG formato stegofailas.



33 pav. Hofmano kodavimo žingsnis

3.1.2. „F5A“ atkodavimas

Pateikiamas „F5A“ steganografinio informacijos išgavimo iš failo algoritmas

3.1.2.1. Algoritmui reikalingi duomenys

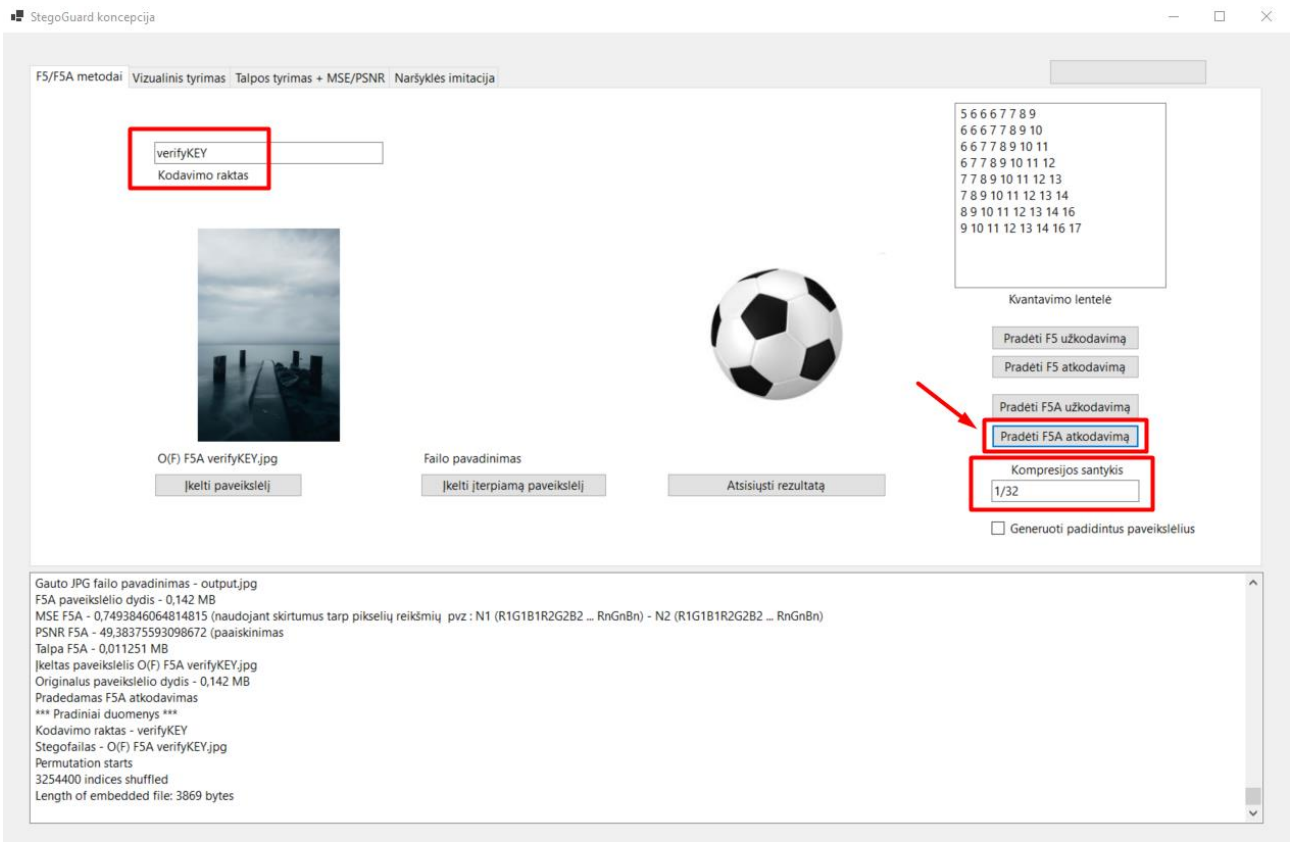
- „F5A“ algoritmu gautas stegofailas
- Kodavimo raktas

3.1.2.2. Hofmano atkodavimas

Hofmano kodavimu taip pat galime ir atgaminti buvusias mūsų buferio reikšmes iš kurių atkodavimo algoritme galime atkurti buvusią informaciją.

Šis žingsnis bus atliekamas esamomis „F5“ algoritmo implementacijos priemonėmis.

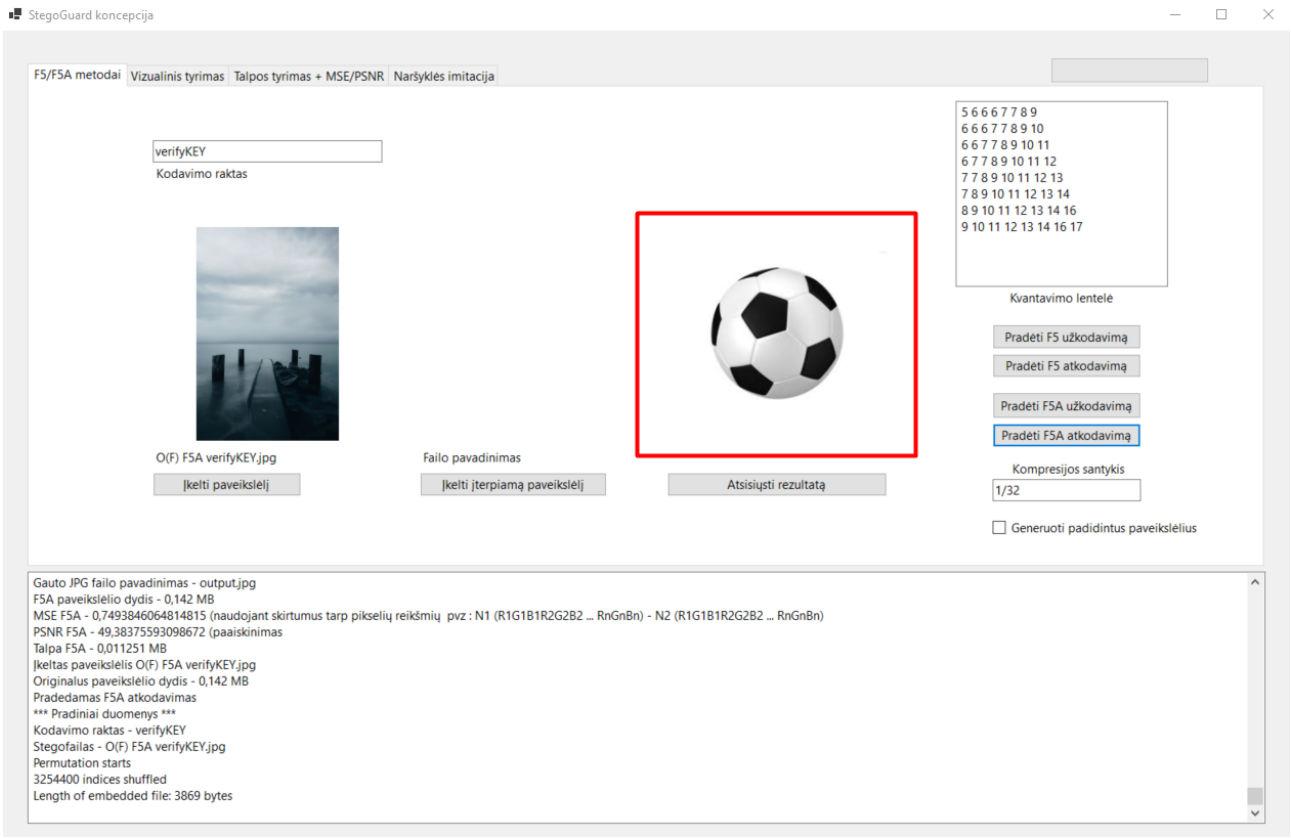
Hofmano kodavimui reikalingas „F5A“ algoritmu gautas stegofailas.



34 pav. F5A atkodavimas

3.1.2.3. Paslėptos informacijos išgavimas pagal raktą

Informacijos išgavimui mums reikalinga žinoti mūsų Kodavimo raktą. Pagal jį vėl yra sugeneruojamas pseudo atsitiktinių skaičių generatorius ir išsiaiškinama, kuriose failo vietose yra užkoduota informacija. Tuomet iš šių vietų yra ištraukiama failo informacija Ir gaunamas tikrasis paveikslėlis.



35 pav. Po atkodavimo gautas rezultatas

3.2. „StegoGuard“ įskiepio aprašymas

Toliau pateikiama informacija įskiepio naudojimui.

3.3. Tipinis įskiepio naudojimas autorinių teisių apsaugai

Naudojami sutartiniai ženklai :

A – įprastas, niekuo neapdorotas paveikslėlis. (Nedalyvaujantis Stego operacijose, bet turi būti rodomas svetainėje)

C – nešyklė, bet koks nesvarbus paveikslėlis.

F – visiems žinomi duomenys, skirti patvirtinimui, jog naudojamas steganografinis algoritmas.

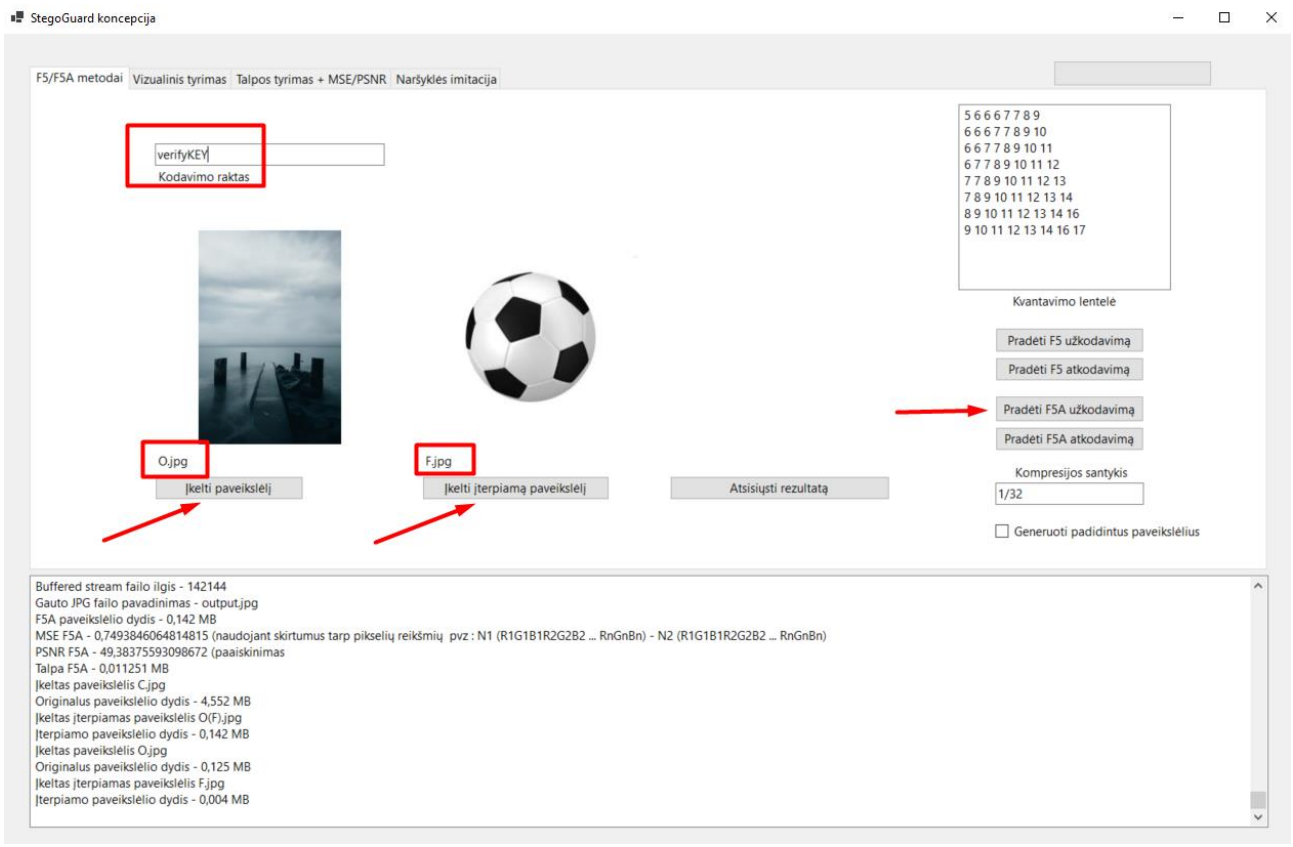
O – tikrasis paveikslėlis, kurį norima apsaugoti nuo neteisėto panaudojimo.

36 pav. Pavyzdžiuose naudojami paveikslėliai

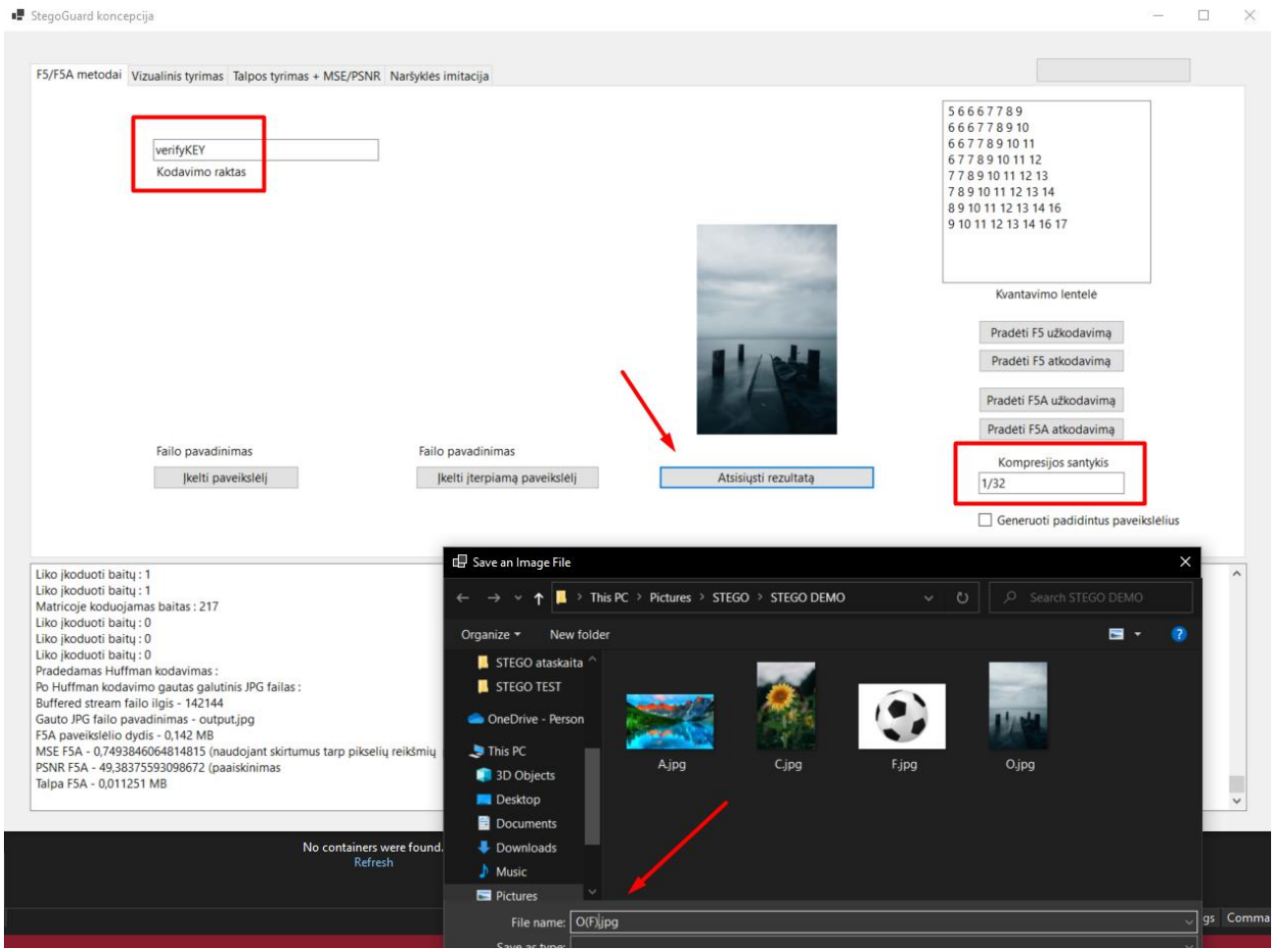
O(F) – į tikrąjį paveikslėlį įterptas patvirtinimo duomenų failas.

C(O(F)) – į nešyklę įterptas tikrasis paveikslėlis, kuriame užkoduotas patvirtinimo duomenų failas.

Pasiruošimas – Autorius užšifruoja visiems žinomą informaciją – paveikslėlį „F“, su visiems žinomu raktu „verifyKEY“. Gauname – O(F).

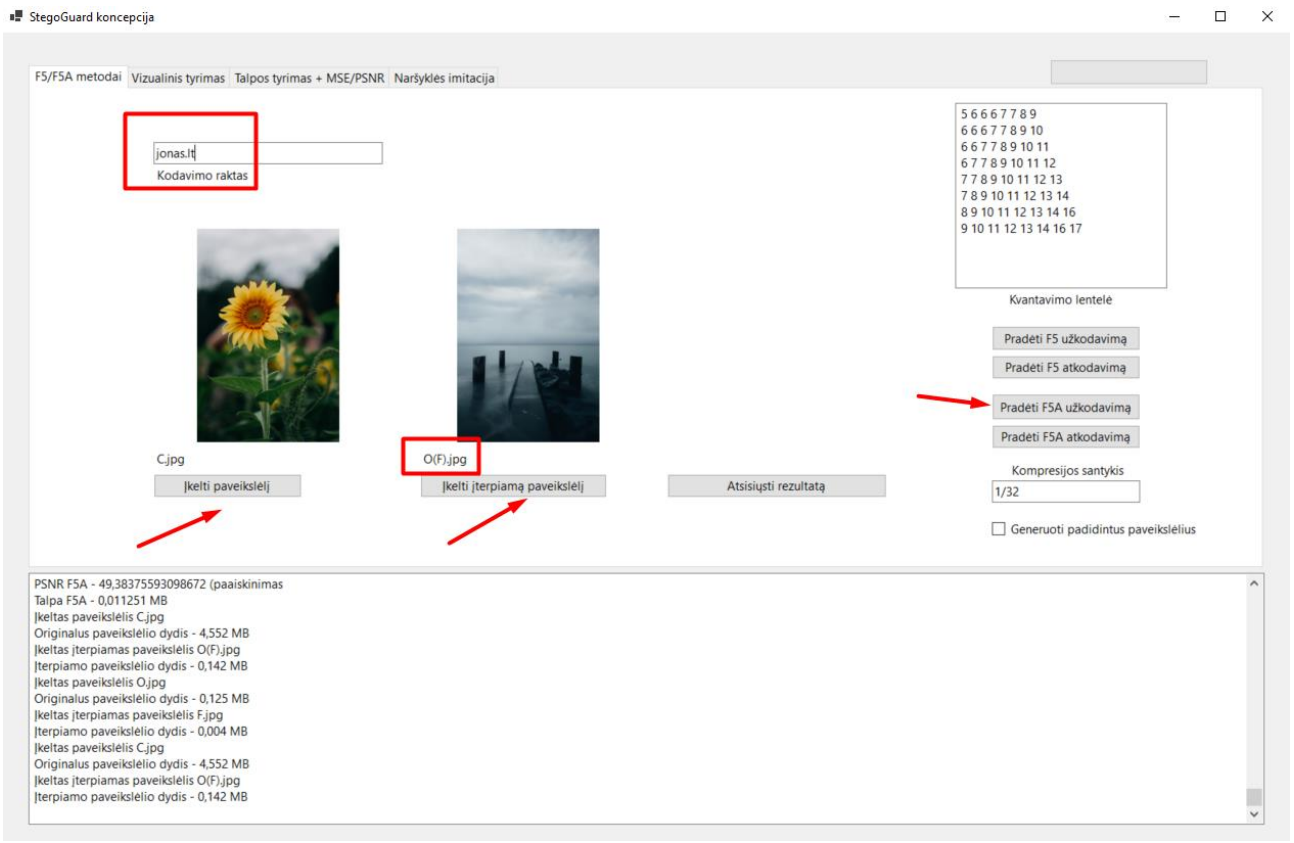


37 pav. Pasiruošimas StegoGuard įskiepio naudojimui

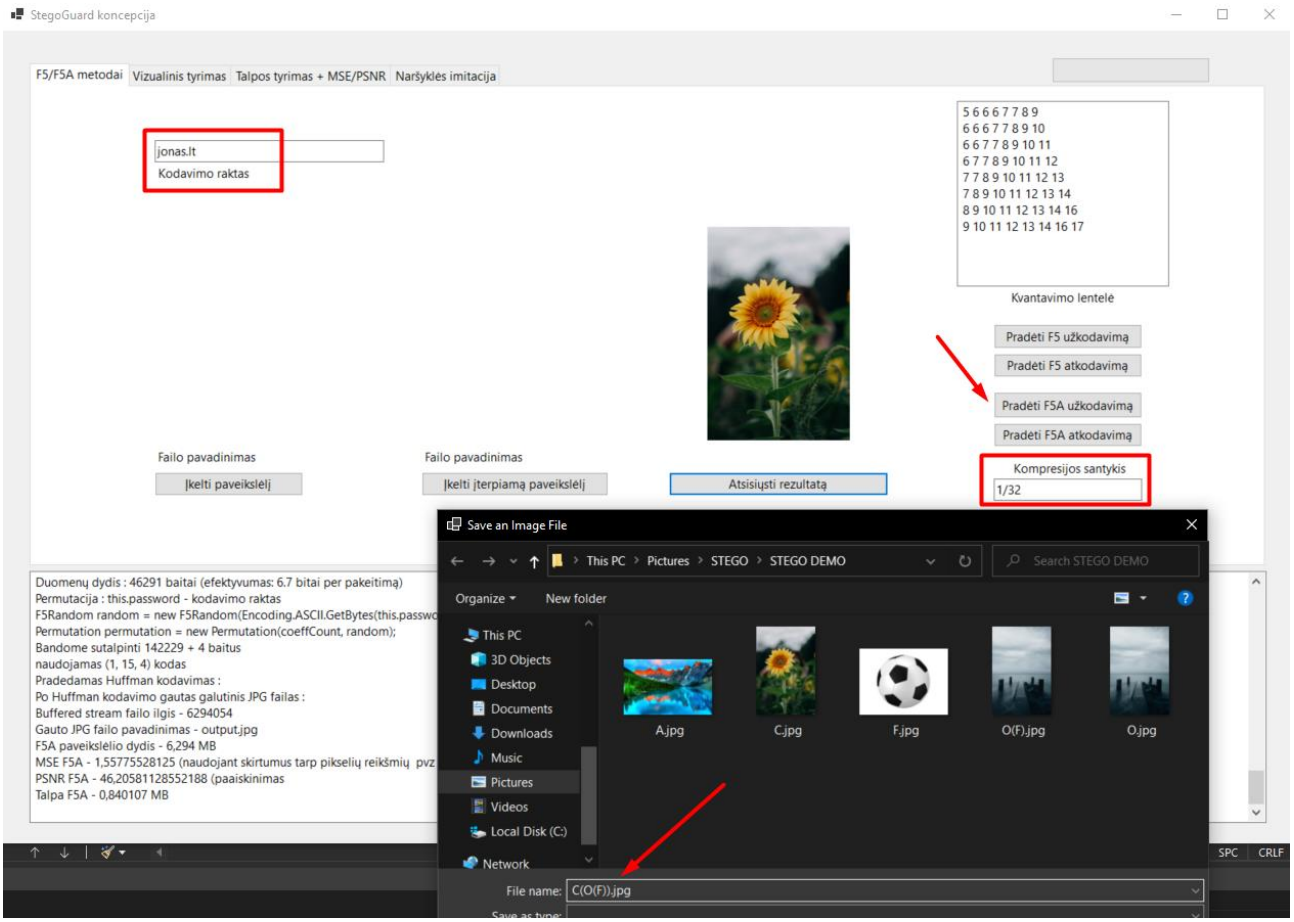


38 pav. Pasiruošimas StegoGuard įskiepio naudojimui

Tuomet į nešyklės failą C įdedame O(F), originalų paveikslėlį su įterptais patvirtinimo duomenimis F. Kaip raktą naudojame domeno pavadinimą, kuriame paveikslėlis galės būti rodomas.



39 pav. Pasiruošimas StegoGuard įskiepio naudojimui



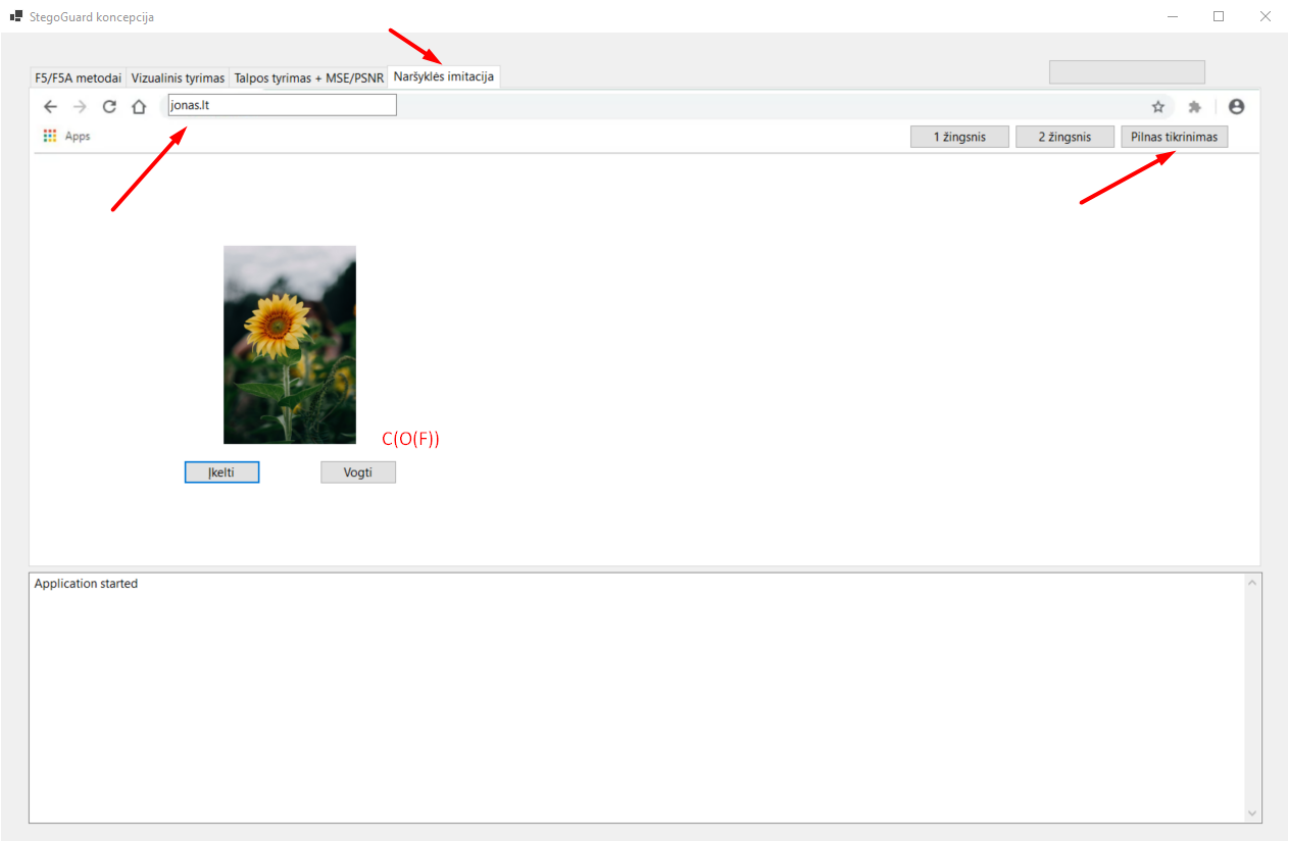
40 pav. Pasiruošimas StegoGuard įskiepio naudojimui

3.3.1. Scenarijus teisėtai naudojamo turinio atveju

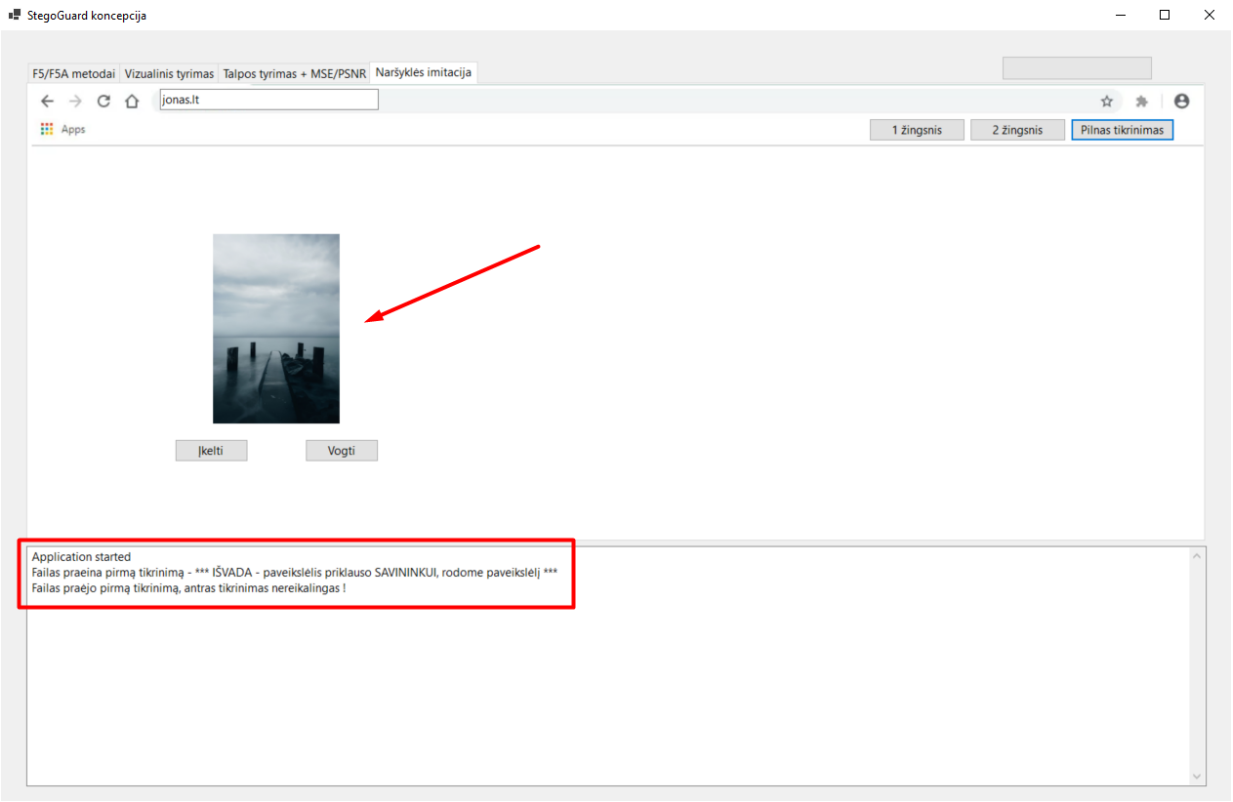
Praktiškai vaizduojamas ir aprašomas scenarijus teisėtai naudojamo turinio atveju.

Įskiepio veikimas pagrįstas dviem tikrinimais – bandymais atšifruoti ir darant išvadas iš gauto rezultato.

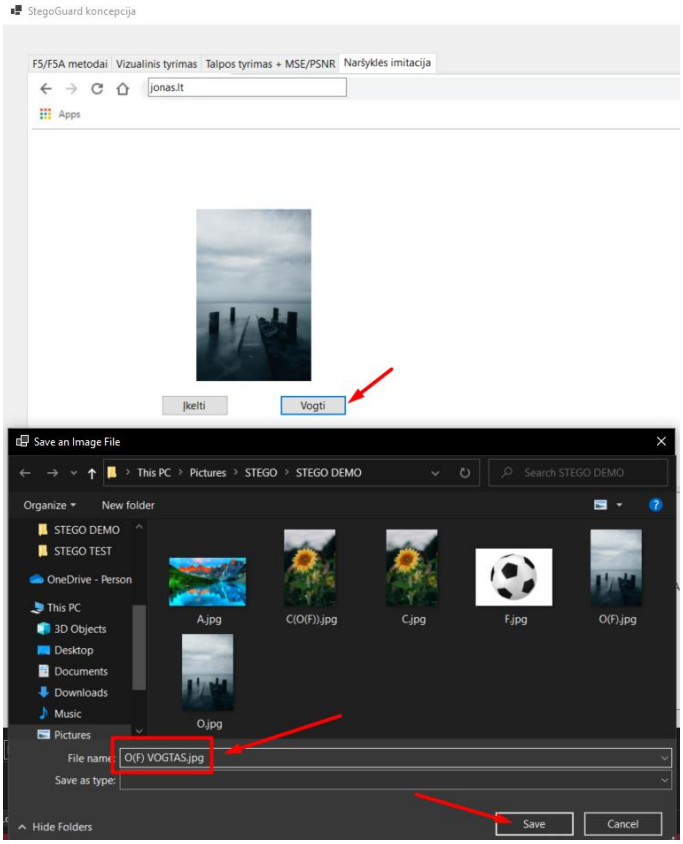
Tikriname – jei galime atlikti atšifravimą pagal domeną ir tuomet pagal raktą verifyKEY, gaunant visiems žinomą paveikslėlį F. Tuomet turinys tikrai yra teisėtas.



41 pav. StegoGuard įskiepio veikimas



42 pav. StegoGuard įskiepio veikimas



43 pav. Paveikslėlio vagystė

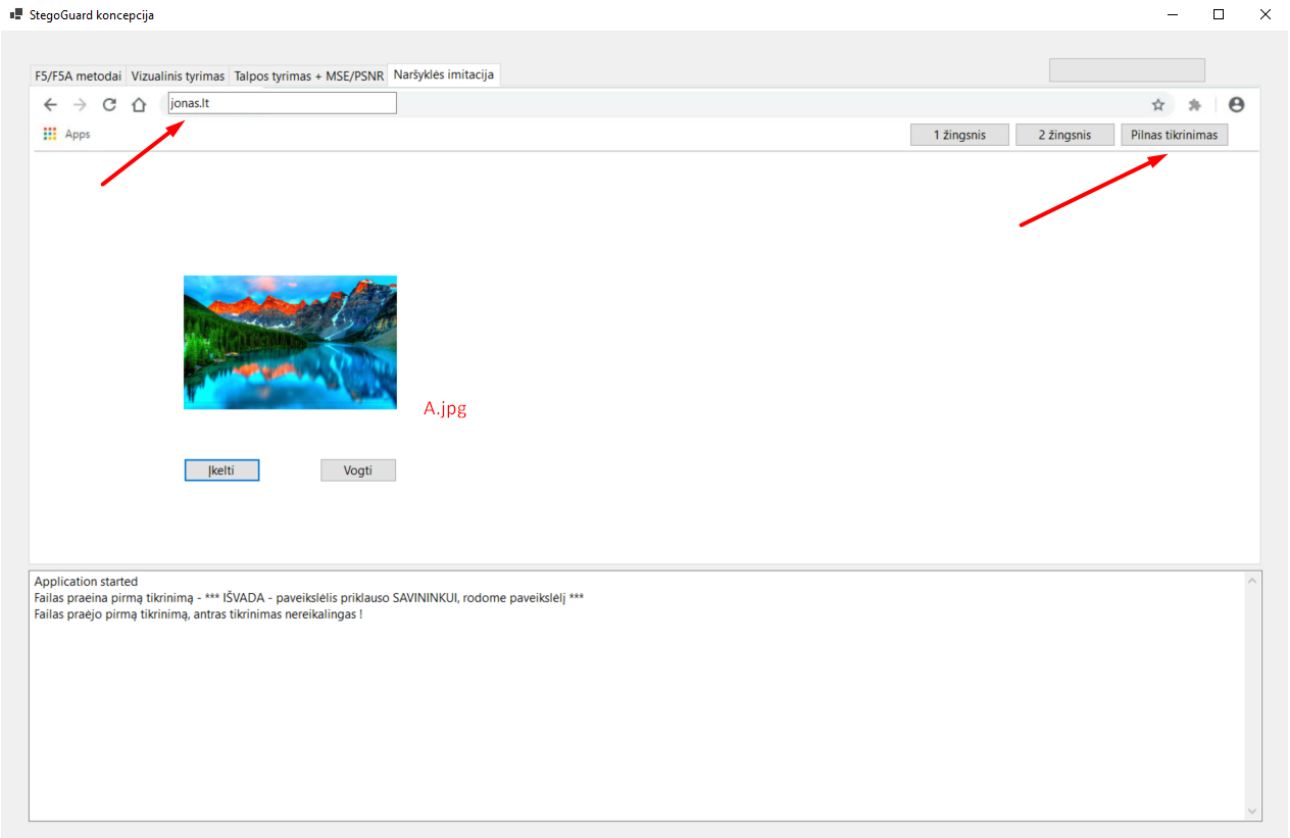
Testavimo metu pastebėta, jog realiu atveju užtenka pirmo tikrinimo, antras atšifravimas yra nebūtinas žingsnis norint užtikrinti, jog paveikslėlis priklauso savininkui, tačiau jis būtinas kitais atvejais, kuriuos apžvelgsime tolimesniuose skyriuose.

Turinys atsišifruoja abiem raktais, todėl yra rodomas, t.y. rodomas O(F). (Atšifruojant domenu)

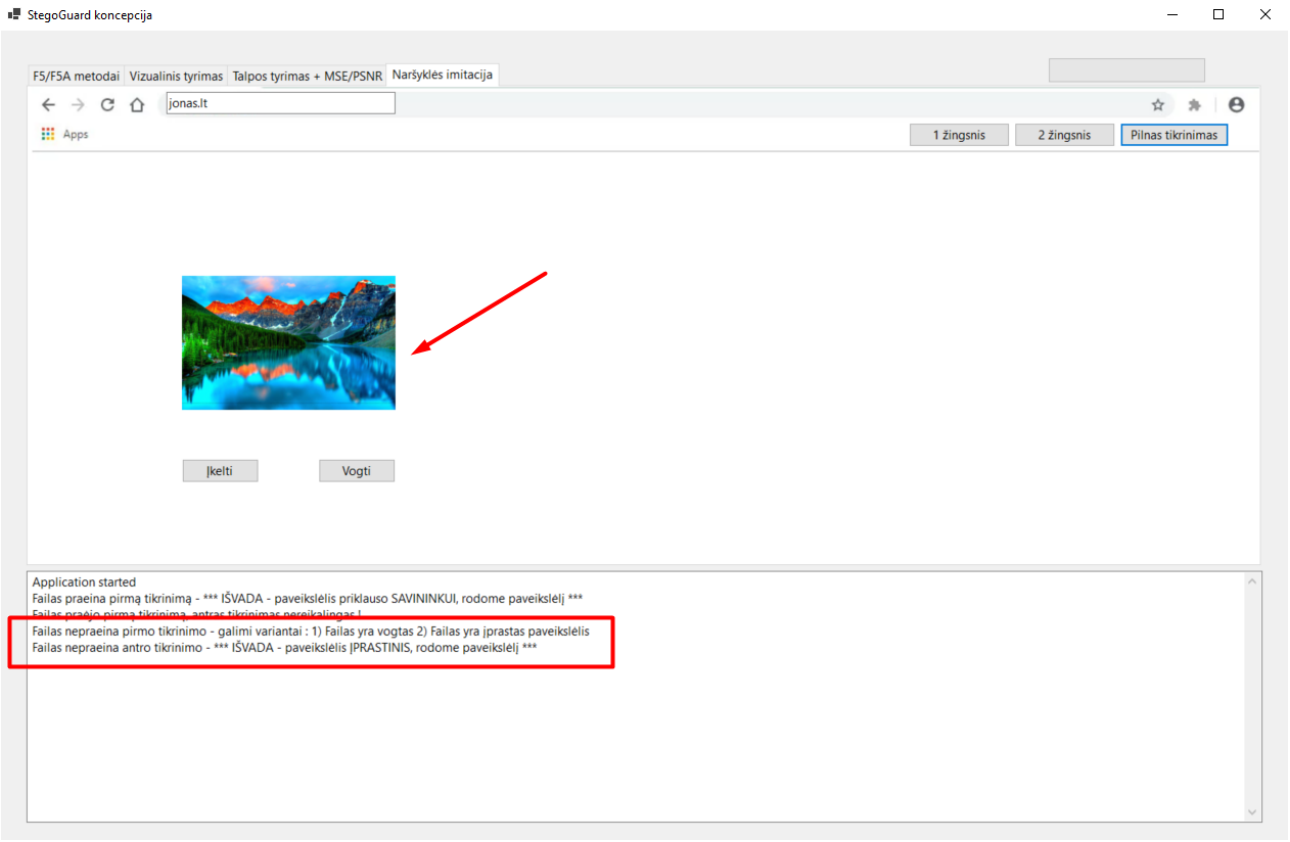
Šį paveikslėlį, „pavagiame“ ir bandysime jį panaudoti kitame scenarijuje.

Taip pat svarbu, jog mūsų svetainėje galėtų būti rodomi ir paprasti, niekuo nešifruoti paveikslėliai, todėl turime patikrinti ir A veikimą.

Jis turi neatsišifruoti, nei vienu iš raktų – domenu, ar verifyKEY.



44 pav. StegoGuard įskiepio veikimas



45 pav. StegoGuard įskiepio veikimas

Failas neatsišifruoja nei vienu iš raktų todėl yra įprastas niekuo neapdirbtas ir turi būti rodomas svetainėje.

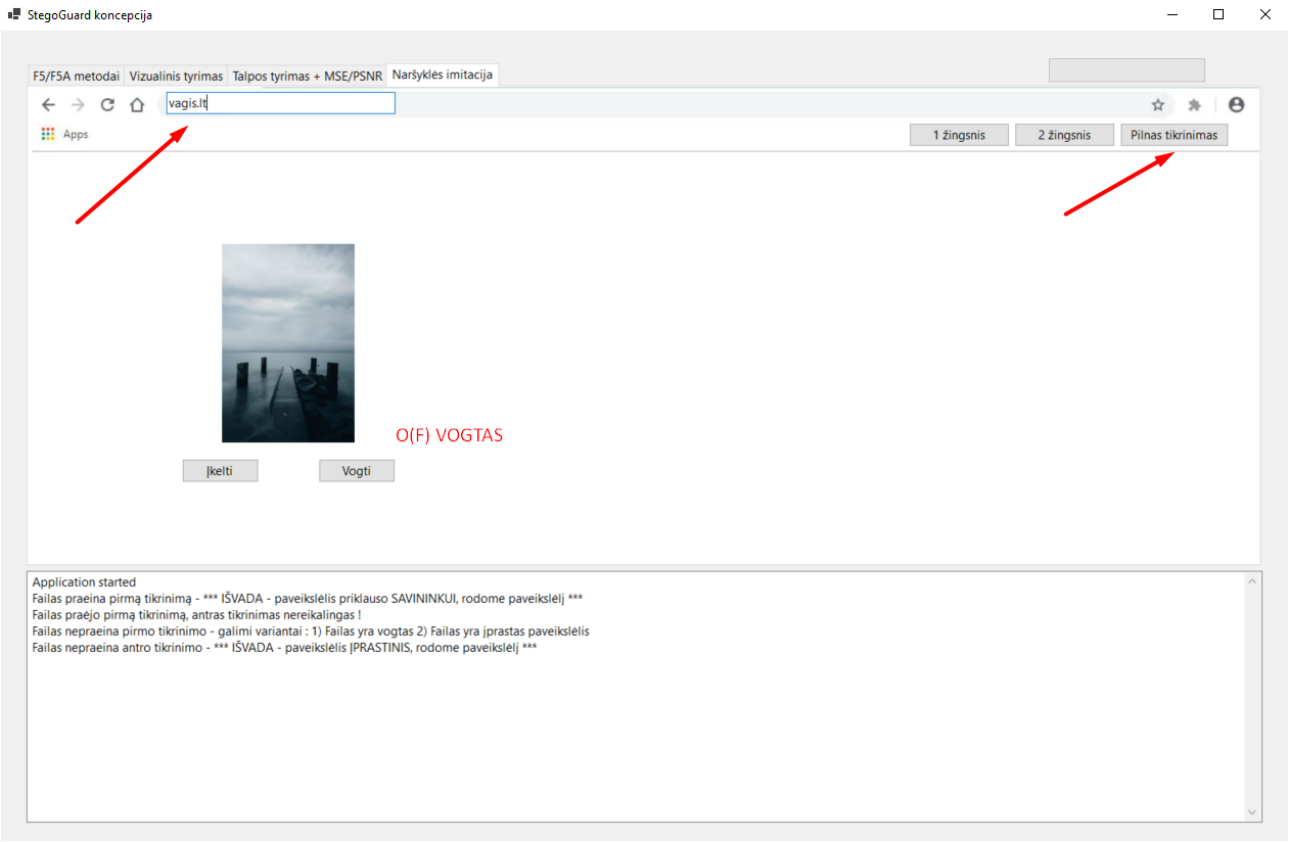
3.3.2. Scenarijus pavogto turinio atveju

Praktiškai vaizduojamas ir aprašomas scenarijus pavogto turinio atveju.

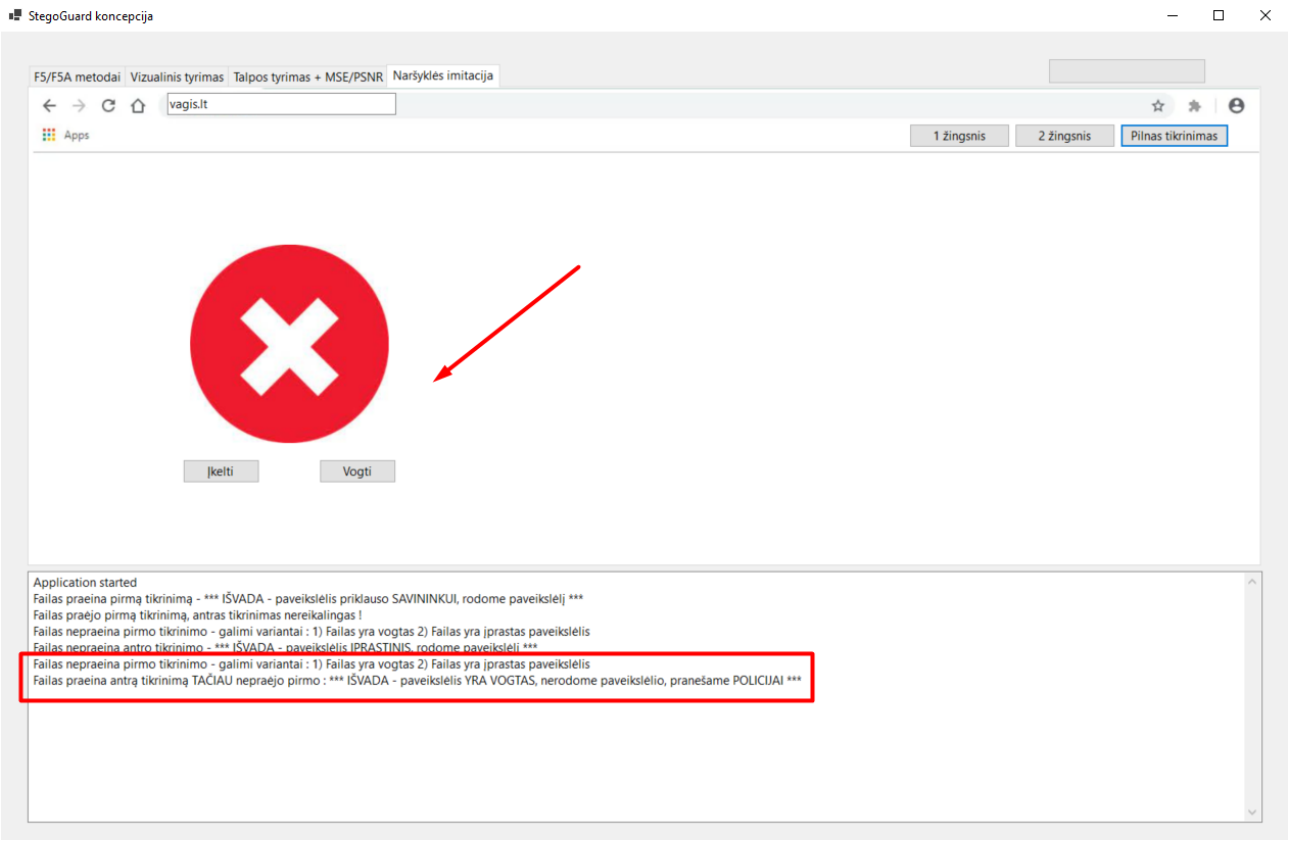
Įskiepio veikimas pagrįstas dviem tikrinimais – bandymais atšifruoti ir darant išvadas iš gauto rezultato.

A.jpg failo veikimas analogiškas pateiktam aukščiau. Niekuo neapdorotas paveikslėlis bus rodomas. (Neatsišifruos nei vienu iš raktų).

Toliau nagrinėjame pavogto paveikslėlio, patalpinto „vagis.lt“ svetainėje atvejį :



46 pav. StegoGuard įskiepio veikimas



47 pav. StegoGuard įskiepio veikimas

Esminis skirtumas nuo niekuo neapdoroto paveikslėlio – jis atsišifruos visiems žinomu raktu verifyKEY. Taip suprasime, jog paveikslėlis iš tiesų yra vogtas.

Išvados ir uždaviniai

1. Pasinaudojant projektinėje dalyje sudarytu algoritmo plano buvo realizuotas naujas steganografinis algoritmas „F5A“, galintis atlikti šias funkcijas:

    - Atpažinti neteisėtai naudojamą turinį, ir pranešti apie tai autorinių teisių savininkui.

    - Apsaugoti atpažintą turinį nuo neteisėto jo panaudojimo.

2. Steganografinis algoritmas „F5A“ panaudotas naujai sukurtame įskiepyje „StegoGuard“ , kurio pagrindinė mintis yra ne tik pranešti autoriui apie jo teisių pažeidimą, tačiau ir aktyviai apsaugoti jo turinį nuo neteisėto panaudojimo. Įgyvendinti du skaitmeninės medijos apsaugos būdai :

    - Integracija į naršyklę, turinio blokavimas (simuliacija)

    - Turinio apsaugojimas raktu

3. Naujas steganografinis metodas „F5A“, kuris remiasi „F5“ algoritmu, praplėsdamas jo talpą naudojant generuojamas kvantavimo lenteles bei pridėdamas naujų savybių, tokių kaip apsauga nuo nešyklės turinio atstatymo bei atkodavimo metu vykdomas pranešimas apie autorinių teisių pažeidimą, kitame skyriuje bus tiriamas bei lyginamas su kitais steganografiniais metodais.

4. Naujai sukurtas autorinių teisių apsaugos įskiepis „StegoGuard“ naudojantis „F5A“ steganografinį algoritmą, bus tiriamas, siekiant nustatyti, ar šis įskiepis sugeba ne tik pastebėti autorinėmis teisėmis apsaugotą turinį, tačiau ir apsaugoti nuo neteisėto jo panaudojimo.

4. Projekto tyrimas

4.1. Tyrimo metodika

Naujo steganografinio metodo F5A projektavimo metu buvo iškelti tokie kriterijai :

Algoritmo talpumas – algoritmas turi leisti paslėpti kuo didesnį informacijos kiekį, nes paveikslėlyje bus slepiama pilna jo kopija, ji turi išlikti kuo geresnės kokybės, todėl slepiamų duomenų kiekis yra ypač svarbus šiam scenarijui.

Algoritmas būti lengvai jungiamas su įterpiamu raktu informacijai užšifruoti – pagrindinis šio scenarijaus kriterijus, nes turinio apsaugojimas raktu yra būtinas norint patikimai apsaugoti paslėptą turinį nuo jo atskleidimo.

Algoritmo tvirtumas - algoritmas turi būti atsparus failo modifikacijoms, tokioms kaip ištampymas, karpymas, suspaudimas ar kitas apdirbimas.

Tyrimo metu bus siekiama išsiaiškinti :

1. Kokią įtaką daro skirtingi kompresijos santykiai naudojami optimizuotose kvantavimo lentelėse ?
2. Kiek optimizuotų kvantavimo lentelių naudojimas gali padidinti steganografinio metodo talpą ?
3. Kokią įtaką optimizuotų kvantavimo lentelių naudojimas daro steganografinio metodo tvirtumui ?
4. Kaip optimizuotų kvantavimo lentelių naudojimas veikia paveikslėlio kokybę ?

4.1.1. Tyrimo metu analizuojamos savybės

Sukurtas „F5A“ steganografinis algoritmas bus lyginamas su „F5“ steganografiniu algoritmu jų palyginimui pasitelkiant išsikeltus kriterijus atitinkančius steganografinių metodų vertinimo būdus:

*** Abu algoritmai naudoja įterpiamą raktą, todėl įterpiamo rakto kriterijus nevertinamas.**

Optimizuotų kvantavimo lentelių naudojimo įtakai nustatyti:

Failo dydžio po steganografinio kodavimo skaičiavimas,

Vidutinės kvadratinės paklaidos skaičiavimas (MSE),

Didžiausio signalo ir triukšmo santykio skaičiavimas (PSNR),

Didžiausio paslėpiamų duomenų kiekio skaičiavimas.

Algoritmų F5 ir F5A tvirtumui nustatyti – Paveikslėlio modifikacijos.

Algoritmų F5 ir F5A vizualiniams skirtumams nustatyti – Steganogramų atvaizdų palyginimas.

4.1.2. Tyrimo rezultatų interpretavimas

Šiame poskyryje pateikiama informacija apie tai kaip suprasti ir vertinti kiekvieno tipo tyrimo rezultatus :

Talpos tyrimo rezultatų interpretavimas :

Apskaičiuojama reikšmė – galimų įterpti bitų į paveikslėlį kiekis. Didesnė reikšmė rodo geresnį rezultatą.

Tvirtumo tyrimo rezultatų interpretavimas :

Karpymas, tampymas, ekrano atvaizdas – Teigiama reikšmė – atlikus modifikaciją atkurti pavyko, Neigiama reikšmė – atlikus modifikaciją atkurti nepavyko.

Vizualinių skirtumų interpretavimas :

Vaizdiniai skirtumai – atsitiktinai pasirinktos paveikslėlio iškarpos vertinimas padidinus vaizdą 1000% vertinant pastebimus pikselių skirtumus tarp steganogramos ir originalo. Didesnė reikšmė rodo prastesnį rezultatą. Taip pat palyginamos paveikslėlių histogramos.

MSE - apskaičiuojama lyginant baitų skirtumus tarp originalaus ir steganografiniu būdu apdirbto failo. Didesnė paklaida gali rodyti didesnį slepiamos informacijos kiekį arba metodo neefektyvumą.

PSNR - santykis tarp didžiausios galios signalo ir triukšmo. Didesnė reikšmę dažniausiai nurodo geresnę steganografiniais metodais apdirbto failo kokybę.

4.2. Tyrimo duomenys

Šiame skyriuje pateikiama tyrimuose naudojamų failų informacija :

Failas	Failo dydis (MB)	
A.jpg	0,469	
F.jpg	0,004	

O.jpg	0,125	
C.jpg	4,552	
O(F) F5.jpg	0,09	
O(F) F5A 1-32.jpg	0,142	

O(F) F5A 8.jpg	0,08	

2 lentelė Tyrimo duomenys

4.3. Optimizuotų kvantavimo lentelių naudojimo įtakos tyrimas

Optimizuotos kvantavimo lentelės gautos tyrimų metu, rodomos šio darbo priede 5.1.

4.3.1. Skaičiavimų rezultatų lentelės

Pateikiamos visų skaičiavimo rezultatų lentelės, pagal kurias yra sudaromos diagramos kituose skyriuose :

Įterpiant į O.jpg, failą F.jpg, taip gaunant O(F).jpg :

Santykis	Failo dydis (MB)	MSE	PSNR	Talpa (MB)
0	0,142	0,749614236	49,38242535	0,011251
1/128	0,142	0,749614236	49,38242535	0,011251
1/64	0,142	0,749614236	49,38242535	0,011251
1/32	0,142	0,749614236	49,38242535	0,011251
1/16	0,142	0,749614236	49,38242535	0,011251
1/8	0,142	0,749614236	49,38242535	0,011251
1/4	0,138	0,756408796	49,3432379	0,011048
1/2	0,131	0,783425926	49,19082421	0,010673
1	0,129	0,785078588	49,18167228	0,01064
2	0,108	0,92879919	48,45158533	0,007958
4	0,095	1,070716435	47,83405892	0,007107
8	0,08	1,627163542	46,01649156	0,005668
16				
32				
64				
128				
F5	0,09	1,287068171	47,0347881	0,006645

48 pav. Tyrimo metu gauti O(F) duomenys

Įterpiant į C.jpg, failą O(F).jpg, taip gaunant C(O(F)).jpg :

Santykis	Failo dydis (MB)	MSE	PSNR	Talpa (MB)
0	6,294	1,557900083	46,2054076	0,840107
1/128	6,294	1,557900083	46,2054076	0,840107
1/64	6,294	1,557900083	46,2054076	0,840107
1/32	6,294	1,557900083	46,2054076	0,840107
1/16	6,294	1,557900083	46,2054076	0,840107
1/8	6,294	1,557900083	46,2054076	0,840107
1/4	6,116	1,593473031	46,10735643	0,824306
1/2	5,903	1,611343021	46,05892359	0,80786
1	5,855	1,655383969	45,94181616	0,807804
2	5,391	1,85209051	45,45418154	0,753531
4	4,562	2,635005458	43,92298842	0,607968
8	3,633	5,505073406	40,72317247	0,440651
16	1,545	18,95084477	35,35451787	0,158492
32				
64				
128				
F5	4,075	4,916776208	41,21399919	0,545752

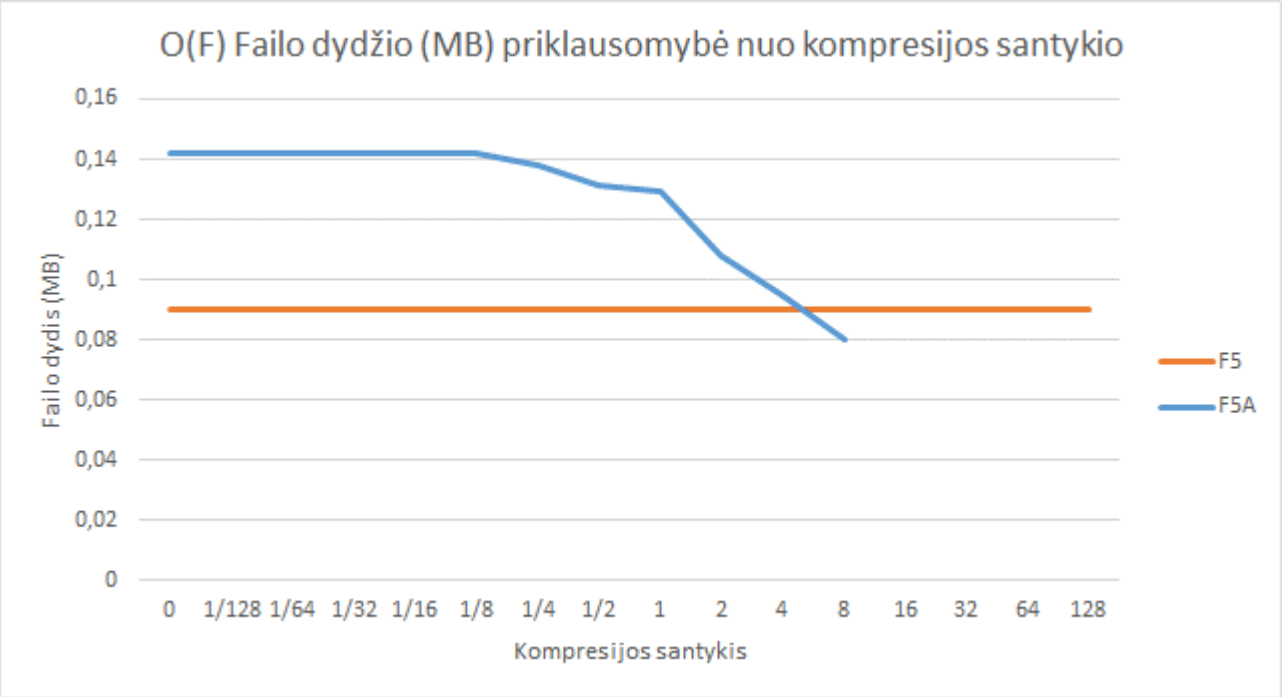
49 pav. Tyrimo metu gauti C(O(F)) duomenys

4.3.2. Failo dydžio po steganografinio kodavimo skaičiavimas

Toliau pateikiama diagrama rodo, kaip prie skirtingo kompresijos santykio, keičiasi steganogramos failo dydis. Steganogramos failo dydis pateikiamas megabaitais (MB) iš failo informacijos.

Diagrama :

O(F) :



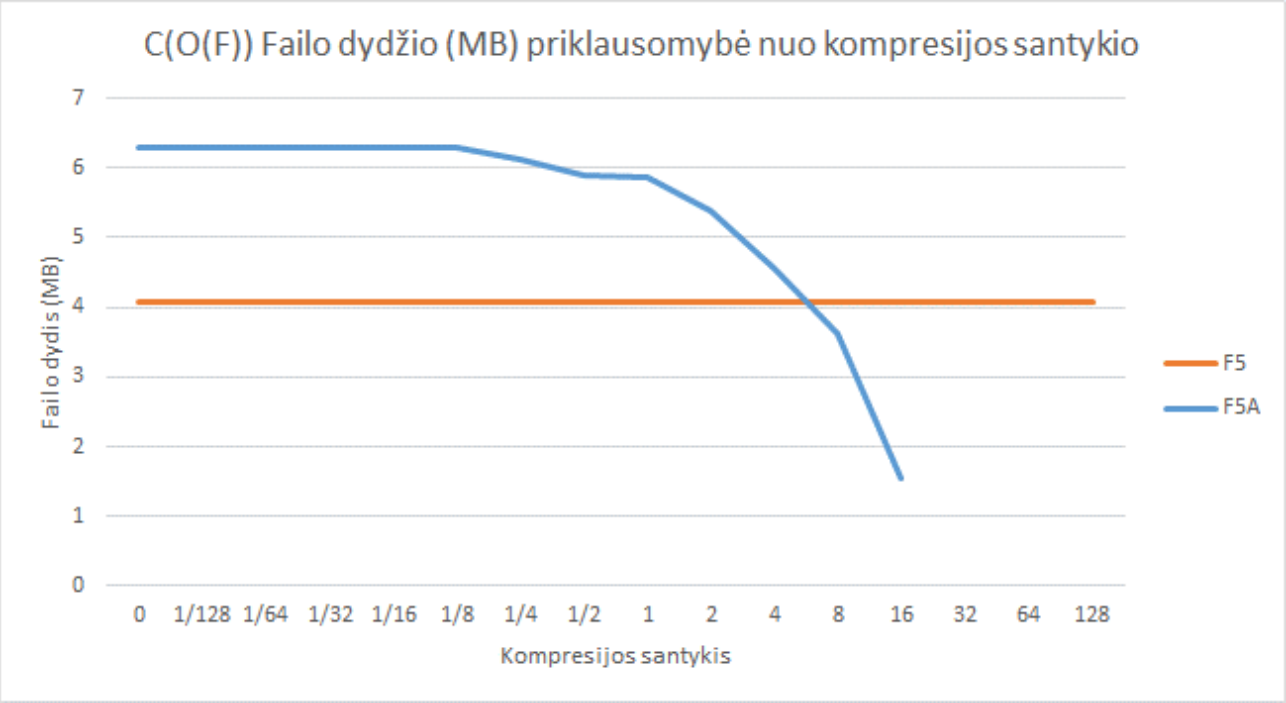
50 pav. O(F) failo dydžio priklausomybė nuo kompresijos santykio

Iš diagramos matome, jog O(F) kodavimo metu didinant kompresijos santykio reikšmę mažėja steganogramos failo dydis. F5A – nuo kompresijos santykio 16 reikšmių nėra, nes užkoduoti nebepavyksta, įterpiamas failas nebetelpa.

Taip pat matome, jog F5 ir F5A kreivės kertasi, todėl egzistuoja toks kompresijos santykis, kuris F5A failo dydį padarytų tokį patį, kaip ir F5. Naudojant kompresijos santykį 8, galima būtų optimizuoti steganografinį algoritmą, kuo mažesniam steganogramos failo dydžiui gauti, taip pagerinant F5 algoritmą.

Matome, jog F5A algoritmas, prie 1/128 kompresijos santykio padidina steganogramos failo dydį, O(F) atveju pastebimas 1.577 karto padidėjęs steganogramos failo dydis, palyginus su F5 algoritmu.

C(O(F)) :



51 pav. C(O(F)) failo dydžio priklausomybė nuo kompresijos santykio

Iš diagramos matome, jog C(O(F)) kodavimo metu F5A – nuo kompresijos santykio 32 reikšmių nėra, nes užkoduoti nebepavyksta, įterpiamas failas nebetelpa.

Koduojant didesnio dydžio paveikslėlį C(O(F)) pavyksta užkoduoti ir su 16 kompresijos santykiu, kuris leidžia dar reikšmingiau sumažinti steganogramos failo dydį.

Matome, jog F5A algoritmas, prie 1/128 kompresijos santykio padidina steganogramos failo dydį, C(O(F)) atveju pastebimas 1.544 karto padidėjęs steganogramos failo dydis, palyginus su F5 algoritmu.

4.3.3. MSE skaičiavimas

Toliau pateikiama diagrama rodo, kaip prie skirtingo kompresijos santykio, keičiasi Vidutinės kvadratinės paklaidos (MSE) reikšmė.

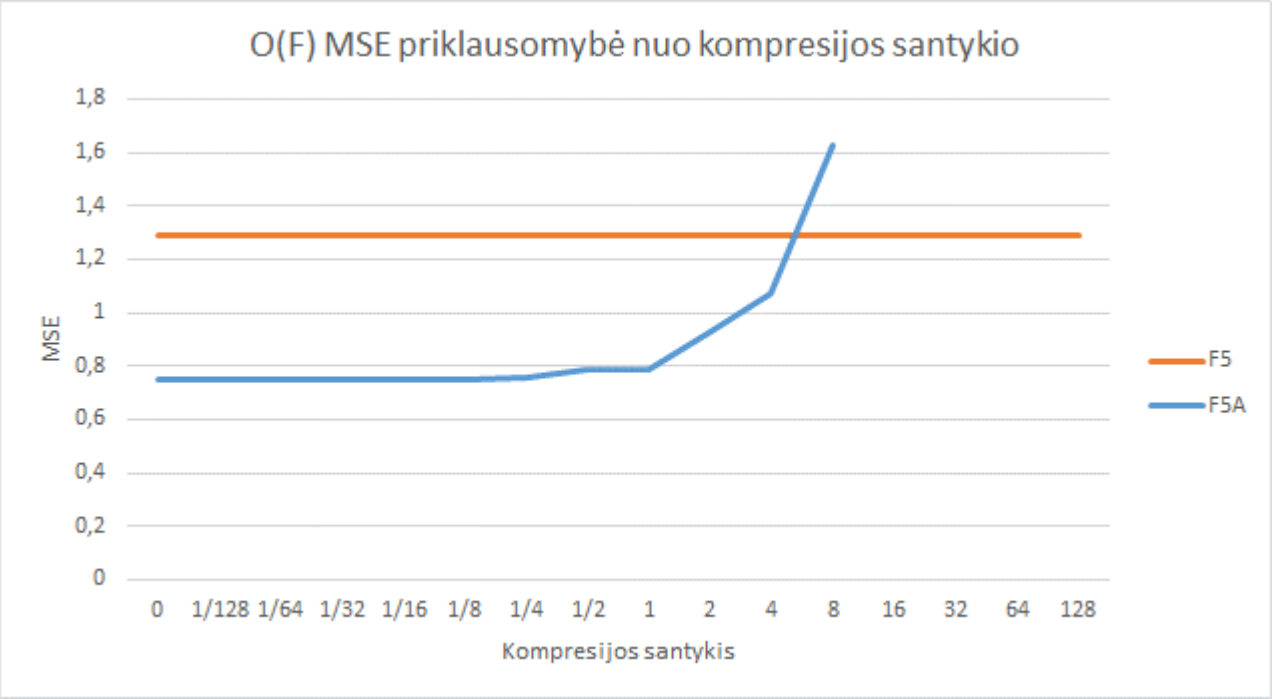
Formulė :

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2$$

Formulėje skirtumas apskaičiuojamas tarp originalaus paveikslėlio pikselio reikšmių (RGB) ir steganogramos paveikslėlio pikselio reikšmių (RGB). Pikselių reikšmės saugomos vienmačiame masyve (reikšmių vektoriuje).

Diagrama :

O(F) :



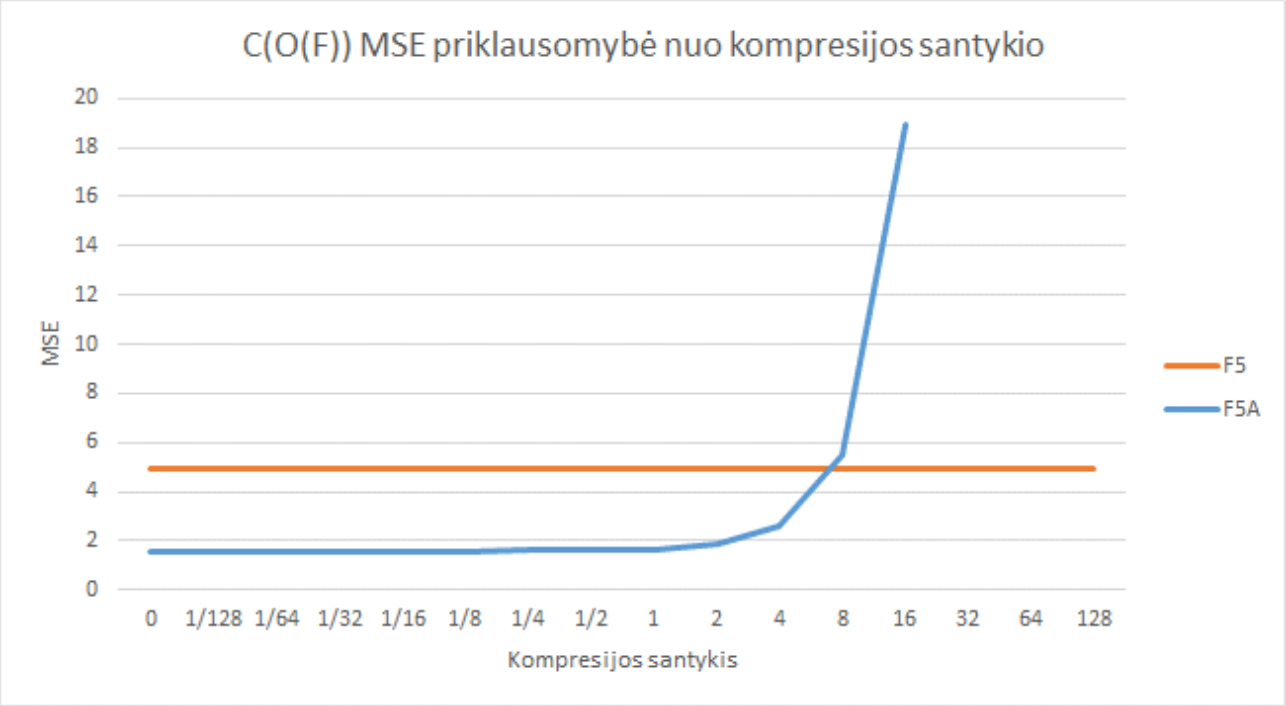
52 pav. O(F) MSE priklausomybė nuo kompresijos santykio

Iš diagramos matome, jog O(F) kodavimo metu didinant kompresijos santykio reikšmę didėja ir MSE reikšmė. F5A – nuo kompresijos santykio 16 reikšmių nėra, nes užkoduoti nebepavyksta, įterpiamas failas nebetelpa.

Didėjanti MSE reikšmė gali reikšti, jog slepiamas didesnis informacijos kiekis arba rodyti metodo neefektyvumą, didesnį neatitikimą su originaliu paveikslėliu. Kadangi tyrimo metu slėpiamos informacijos kiekis visais atvejais buvo vienodas, galime daryti išvadą, jog didėjant kompresijos santykiui, didėja ir vidutinė kvadratinė paklaida (MSE), metodas tampa mažiau efektyvus. Tačiau, taip pat pastebima, jog pagal F5 metodo, naudojančio standartinę JPG kvantavimo lentelę, kreivę jo efektyvumas yra mažesnis nei naudojant F5A metodą su mažesniu nei 4 kompresijos santykiu.

Matome, jog F5A algoritmas, prie 1/128 kompresijos santykio gali sėkmingai padidinti steganogramos efektyvumą, O(F) atveju pastebimas 1.716 karto padidėjęs steganogramos efektyvumas pagal MSE reikšmę, palyginus su F5 algoritmu.

C(O(F)) :



53 pav. C(O(F)) MSE priklausomybė nuo kompresijos santykio

Iš diagramos matome, jog C(O(F)) kodavimo metu F5A – nuo kompresijos santykio 32 reikšmių nėra, nes užkoduoti nebepavyksta, įterpiamas failas nebetelpa.

Koduojant didesnio dydžio paveikslėlį C(O(F)) pavyksta užkoduoti ir su 16 kompresijos santykiu, tačiau toks kompresijos santykis, nors ir sugeba labai reikšmingai sumažinti galutinį steganogramos failo dydį, jis taip pat padaro steganogramą neefektyvią, su dideliu neatitikimu tarp steganogramos ir originalaus paveikslėlio.

Matome, jog F5A algoritmas, prie 1/128 kompresijos santykio gali sėkmingai padidinti steganogramos efektyvumą, C(O(F)) atveju pastebimas 3.156 karto padidėjęs steganogramos efektyvumas pagal MSE reikšmę, palyginus su F5 algoritmu.

4.3.4. PSNR skaičiavimas

Toliau pateikiama diagrama rodo, kaip prie skirtingo kompresijos santykio, keičiasi santykio tarp didžiausios galios signalo ir triukšmo (PSNR) reikšmė.

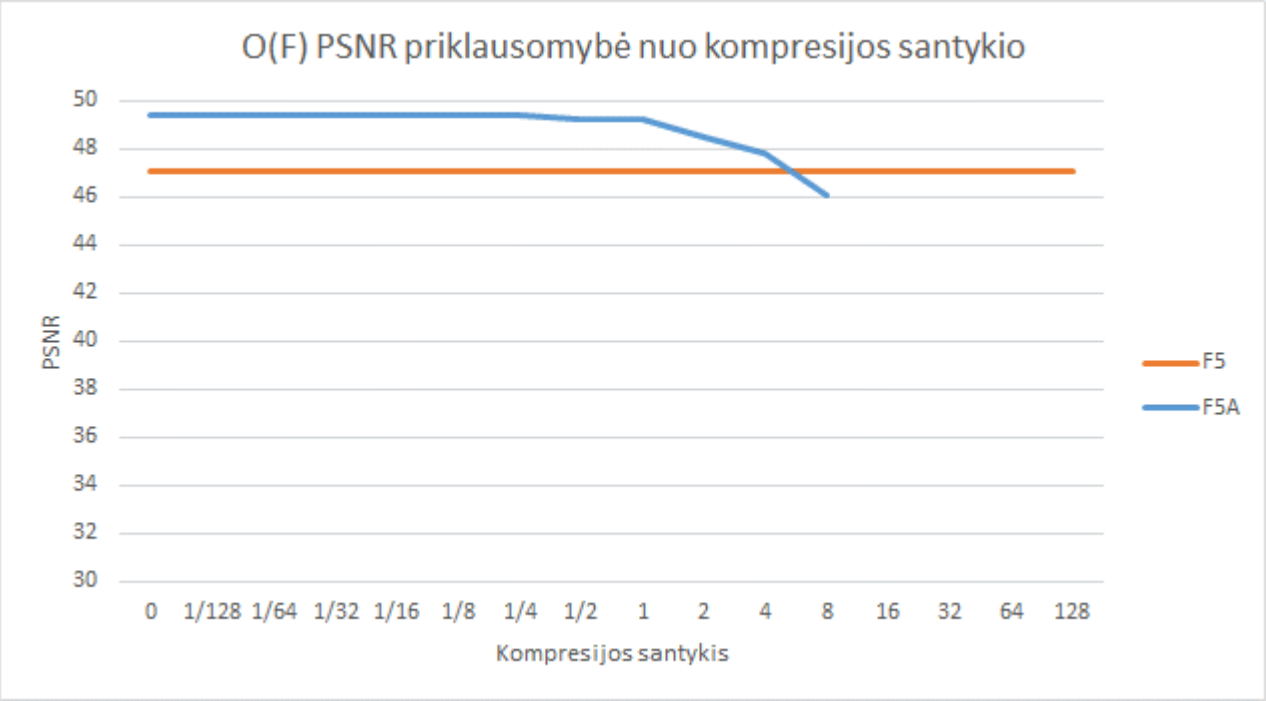
Formulė :

$$PSNR = 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right)$$

Formulėje naudojama prieš tai gauta MSE reikšmė bei maksimali vieno spalvos kanalo pikselio reikšmė spalvotam failui MAX = 255.

Diagrama :

O(F) :



54 pav. O(F) PSNR priklausomybė nuo kompresijos santykio

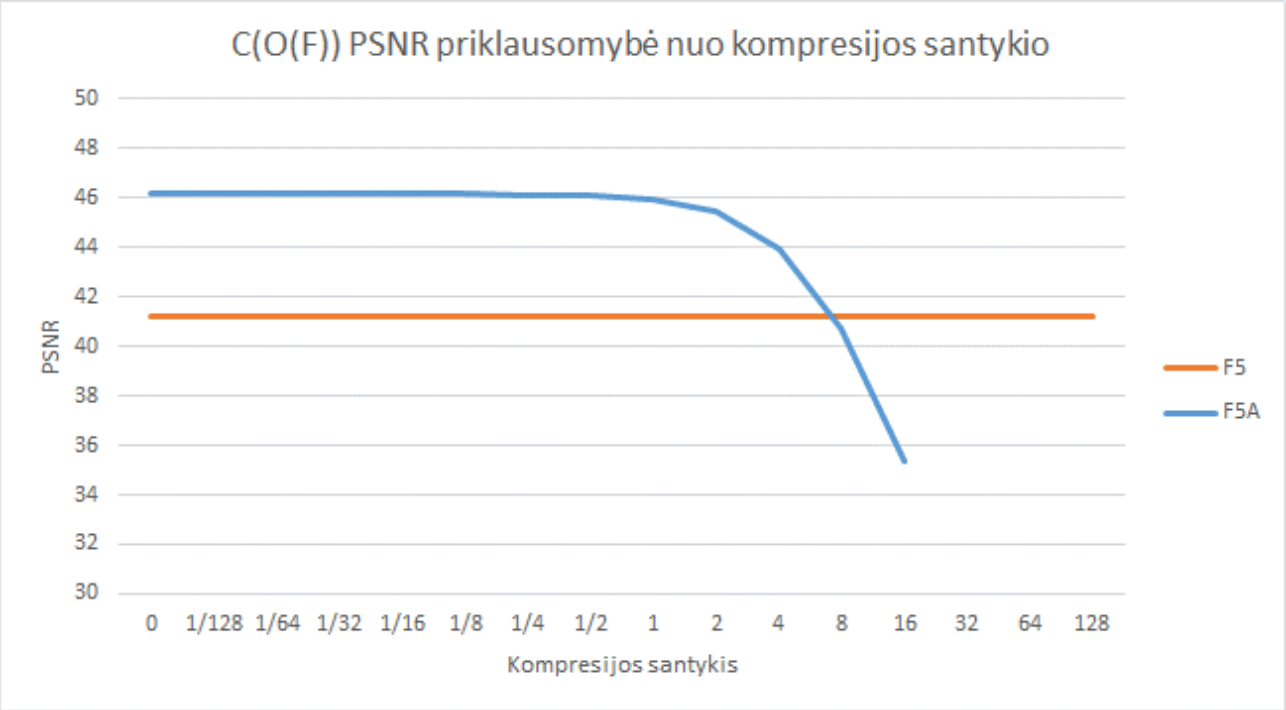
Šiai diagramai specialiai buvo parinktos 30 – 50 PSNR reikšmių ašies minimali ir maksimali reikšmės, siekiant parodyti, koks yra PSNR reikšmių atitrūkimas tarp O(F) ir didesnio paveikslėlio C(O(F)).

Iš diagramos matome, jog O(F) kodavimo metu didinant kompresijos santykio reikšmę mažėja santykis tarp didžiausios galios signalo ir triukšmo (PSNR). F5A – nuo kompresijos santykio 16 reikšmių nėra, nes užkoduoti nebepavyksta, įterpiamas failas nebetelpa.

Didesnė PSNR reikšmė dažniausiai rodo geresnę steganografiniais metodais apdirbto failo kokybę. Tipiniam JPG formato failui įprastos reikšmės yra tarp 30 ir 50 dB PSNR, taip pat reikšmės virš 40 dB reiškia labai gerą paveikslėlio kokybę. Todėl nors ir matomas skirtumas tarp F5 ir F5A metodų, jie abu patenka į labai geros paveikslėlio kokybės rėžius.

Matome, jog F5A algoritmas, prie 1/128 kompresijos santykio gali sėkmingai pagerinti paveikslėlio kokybę, O(F) atveju pastebima 1.121 karto pagerėjusi paveikslėlio kokybė pagal PSNR reikšmę, palyginus su F5 algoritmu.

C(O(F)) :



55 pav. C(O(F)) PSNR priklausomybė nuo kompresijos santykio

Iš diagramos matome, jog C(O(F)) kodavimo metu F5A – nuo kompresijos santykio 32 reikšmių nėra, nes užkoduoti nebepavyksta, įterpiamas failas nebetelpa.

Koduojant didesnio dydžio paveikslėlį C(O(F)) pavyksta užkoduoti ir su 16 kompresijos santykiu. Taip pat palyginus diagramą su O(F) matome, jog koduojant didesnio dydžio paveikslėlį C(O(F)) PSNR reikšmės F5 ir F5A metodams yra santykinai mažesnės taip pat matomas didesnis atitrūkimas tarp F5 ir F5A metodų. Kompresijos santykiui pasiekus reikšmę 16, paveikslėlio kokybė nebepriskiriama prie labai geros kokybės, nes reikšmė nėra didesnė nei 40dB.

Matome, jog F5A algoritmas, prie 1/128 kompresijos santykio gali pagerinti paveikslėlio kokybę, C(O(F)) atveju pastebima 1.121 karto pagerėjusi paveikslėlio kokybė pagal PSNR reikšmę, palyginus su F5 algoritmu.

4.3.5. Talpos skaičiavimas

Toliau pateikiama diagrama rodo, kaip prie skirtingo kompresijos santykio, keičiasi paslėpiamų duomenų kiekis. Paslėpiamų duomenų kiekio skaičiavimas yra F5 bei F5A algoritmo dalis. Jie yra apskaičiuojami pačiame F5 algoritme.

Formulė išreikšta pseudo kodu :

```
int _one = 0; int _large = 0;

for (i = 0; i < coeffCount; i++)
{
    if (i % 64 == 0)
```

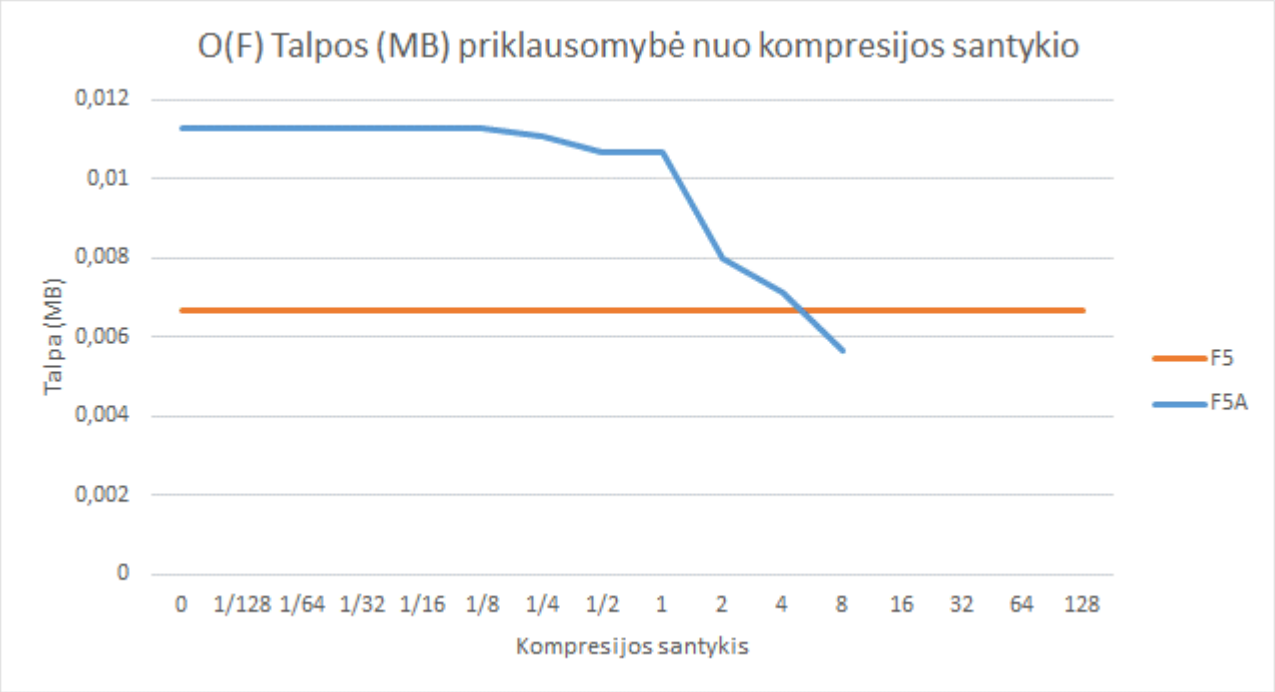
```
            continue;
        else if (coeff[i] == 1 || coeff[i] == -1)
            _one++;
        else if (coeff[i] == 0)
            _zero++;
    }
    _large = coeffCount - _zero - _one - coeffCount / 64;

    TALPA = _large + (int)(0.49 * _one);
```

Formulėje coeffCount – DCT koeficientų kiekis, coeff – koeficientų masyvas. TALPA – apskaičiuota talpa bitais.

Diagrama :

O(F) :



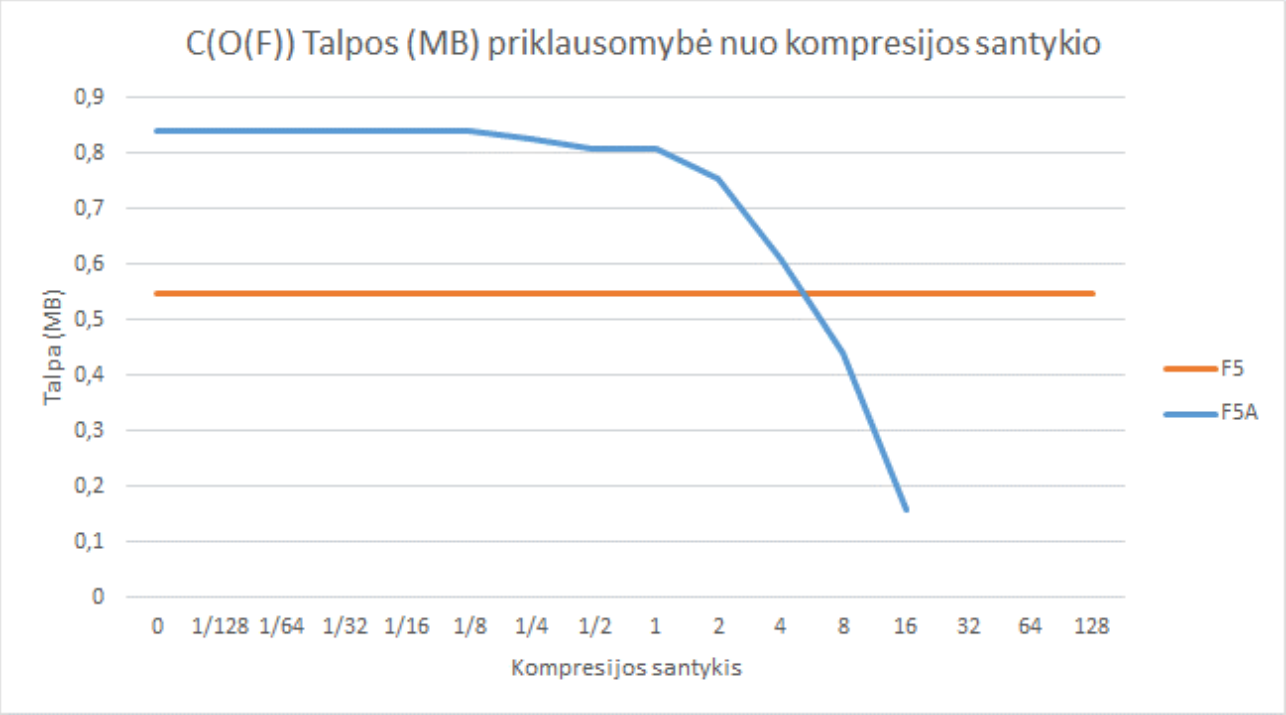
56 pav. O(F) talpos priklausomybė nuo kompresijos santykio

Iš diagramos matome, jog O(F) kodavimo metu didinant kompresijos santykio reikšmę steganografinio algoritmo talpa. F5A – nuo kompresijos santykio 16 reikšmių nėra, nes užkoduoti nebepavyksta, įterpiamas failas nebetelpa.

Talpos kriterijus yra svarbiausias kriterijus mūsų algoritmui, nes F5A algoritmas yra skirtas stegoguard įskiepiui, kuris į vieną paveikslėlį turi sutalpinti kitą paveikslėlį, kuriame taip pat bus užkoduota informacija.

Matome, jog F5A algoritmas, prie 1/128 kompresijos santykio gali sėkmingai padidinti steganografinio algoritmo talpą, O(F) atveju pastebimas 1.693 karto padidėjęs algoritmo talpumas, palyginus su F5 algoritmu.

C(O(F)) :



57 pav. C(O(F)) talpos priklausomybė nuo kompresijos santykio

Iš diagramos matome, jog C(O(F)) kodavimo metu F5A – nuo kompresijos santykio 32 reikšmių nėra, nes užkoduoti nebepavyksta, įterpiamas failas nebetelpa.

Koduojant didesnio dydžio paveikslėlį C(O(F)) pavyksta užkoduoti ir su 16 kompresijos santykiu. Tačiau tokiu atveju labai sumažėja steganografinio algoritmo talpa. Didesnio dydžio paveikslėlio kodavimo C(O(F)) atveju, taip pat pastebimas talpos padidėjimas.

Matome, jog F5A algoritmas, prie 1/128 kompresijos santykio taip pat gali sėkmingai padidinti steganografinio algoritmo talpą, C(O(F)) atveju pastebimas 1.539 karto padidėjęs algoritmo talpumas, palyginus su F5 algoritmu.

4.4. „F5A“ steganografinio metodo tvirtumo tyrimas

Tyrimo metu vykdomi įvairūs steganogramos pakeitimai, bandoma failą apkarpyti, ištempti arba atlikti ekrano atvaizdą. Atliekant pakeitimus, fiksuojamas atlikto pakeitimo dydis bei bandoma išgauti slaptą informaciją iš paveikslėlio. Fiksuojamas bandymo rezultatas.

Failas	Modifikacija	Pakeitimo dydis	F5 rezultatas	F5A rezultatas
O(F).jpg	Karpymas	- 10px * 10px	Atkurti nepavyko	Atkurti nepavyko
O(F).jpg	Tampymas	+ 1% plotis	Atkurti nepavyko	Atkurti nepavyko
O(F).jpg	Ekrano atvaizdas	-	Atkurti nepavyko	Atkurti nepavyko
C(O(F)).jpg	Karpymas	- 10px * 10px	Atkurti nepavyko	Atkurti nepavyko
C(O(F)).jpg	Tampymas	+ 1% plotis	Atkurti nepavyko	Atkurti nepavyko
C(O(F)).jpg	Ekrano atvaizdas	-	Atkurti nepavyko	Atkurti nepavyko

58 pav. Failo modifikacijų tyrimo duomenys

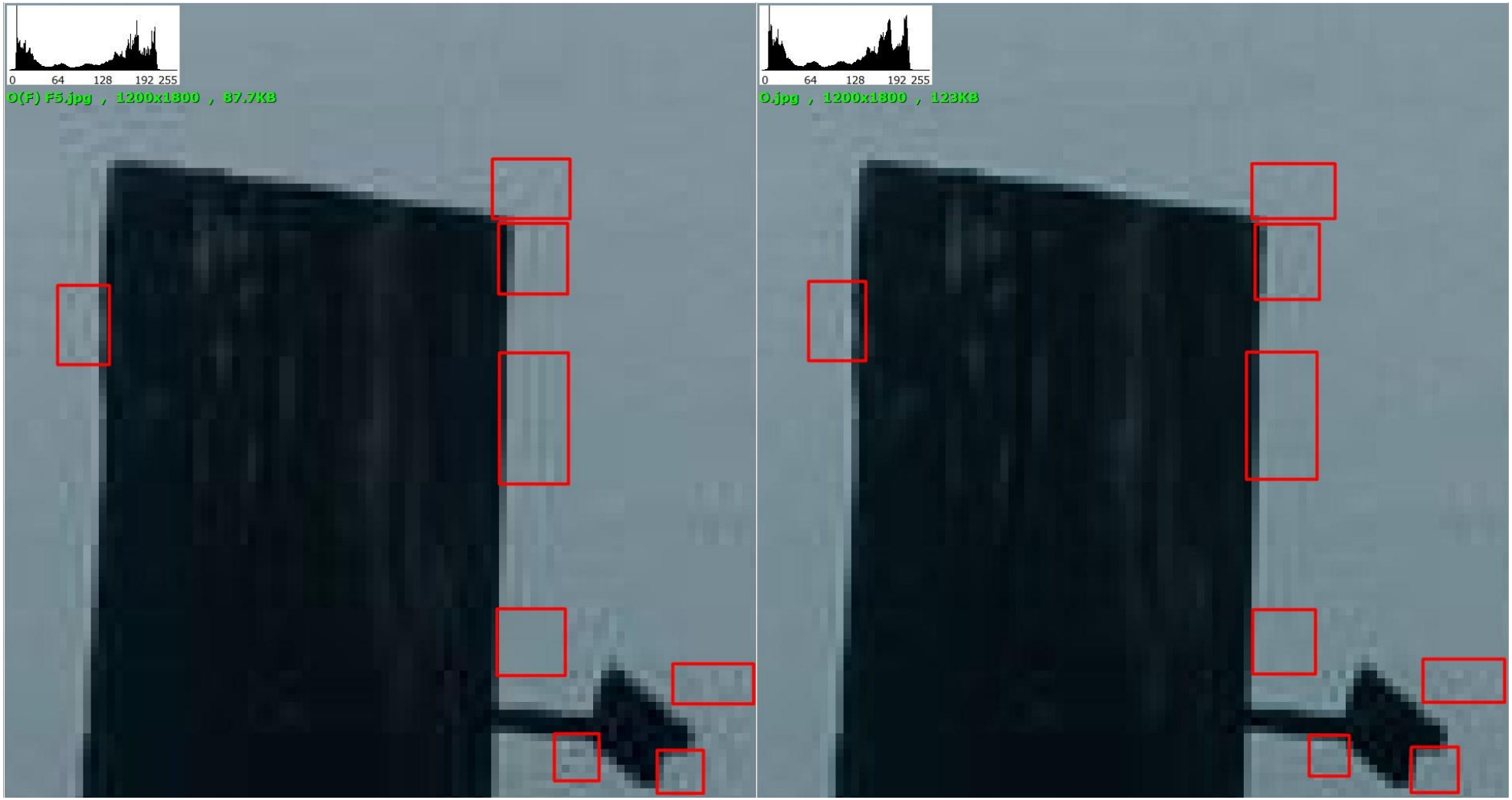
Tyrimo metu buvo stengiamasi atitikti realaus pasaulio pavyzdį, kuomet vartotojai dažniausiai atlieka modifikacijas JPEG formato failams (steganogramoms) programose tokiose kaip MS Paint ar Adobe Photoshop.

Atlikus tyrimą, pastebėta, jog F5 ir F5A steganografinis metodai nėra atsparūs paveikslėlio modifikacijoms. Modifikacijų metu atlikus karpymą, tampymą ar ekrano atvaizdą paveikslėlio duomenys buvo antrą kartą suspaudžiami JPEG formatu, todėl įterpta slapta informacija buvo sugadinama.

4.5. „F5A“ steganografinio metodo vizualinis tyrimas

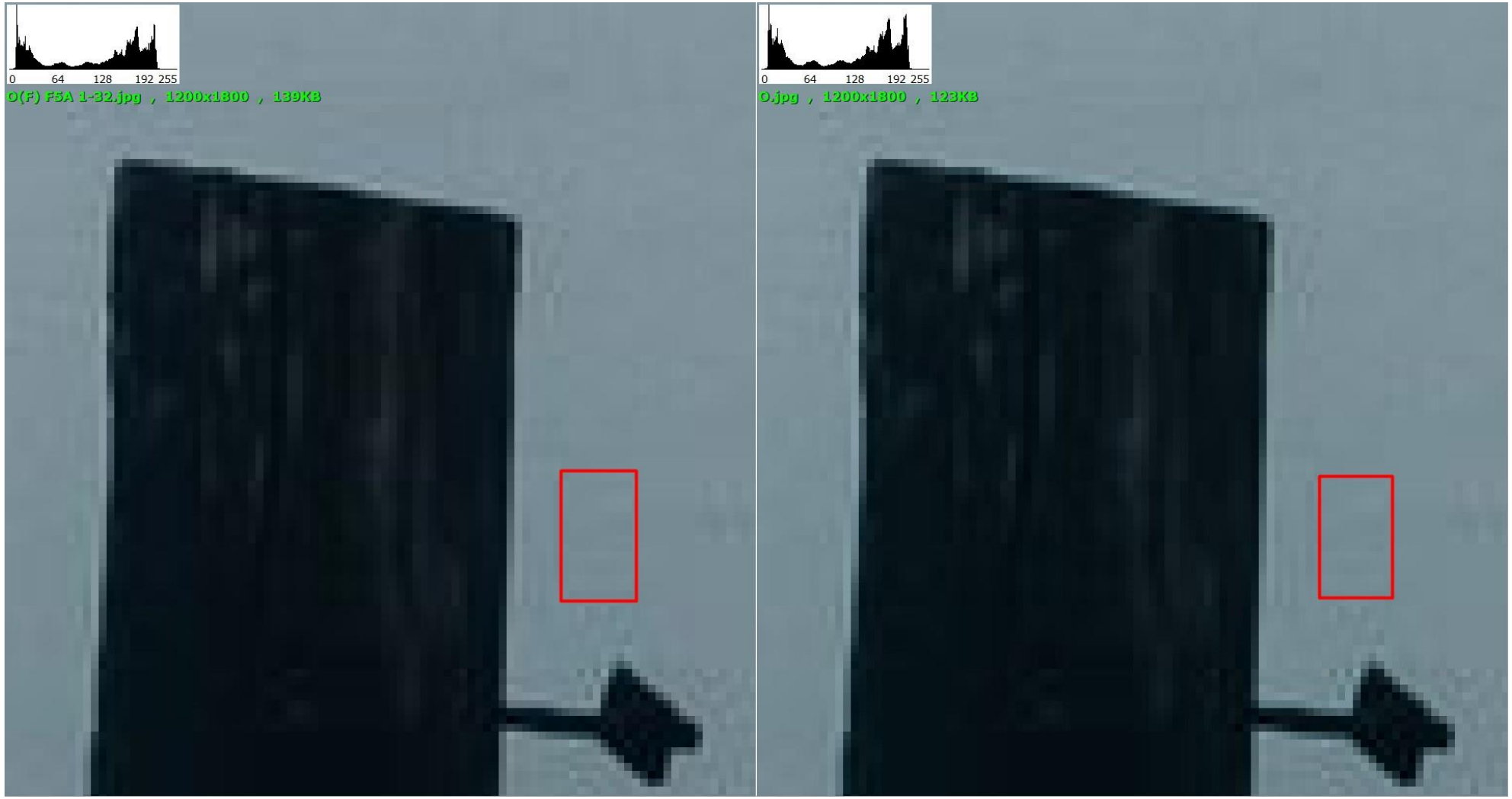
Tyrimo metu pasirenkama steganogramos dalis padidinama 1000% ir ieškoma neatitikimų su ta pačia originalaus paveikslėlio dalimi. Radus neatitikimus pikseliuose, jie yra skaičiuojami. Plika akimi matomi neatitikimai apibraukiami raudonu kvadratu. Didesnis neatitikimų skaičius nurodo prastesnį rezultatą. Taip pat palyginamos paveikslėlių histogramos. Tyrimui naudojama programinė įranga – „FastStone Image Viewer 7.7“

O(F) F5.jpg ir O.jpg: (1000%)



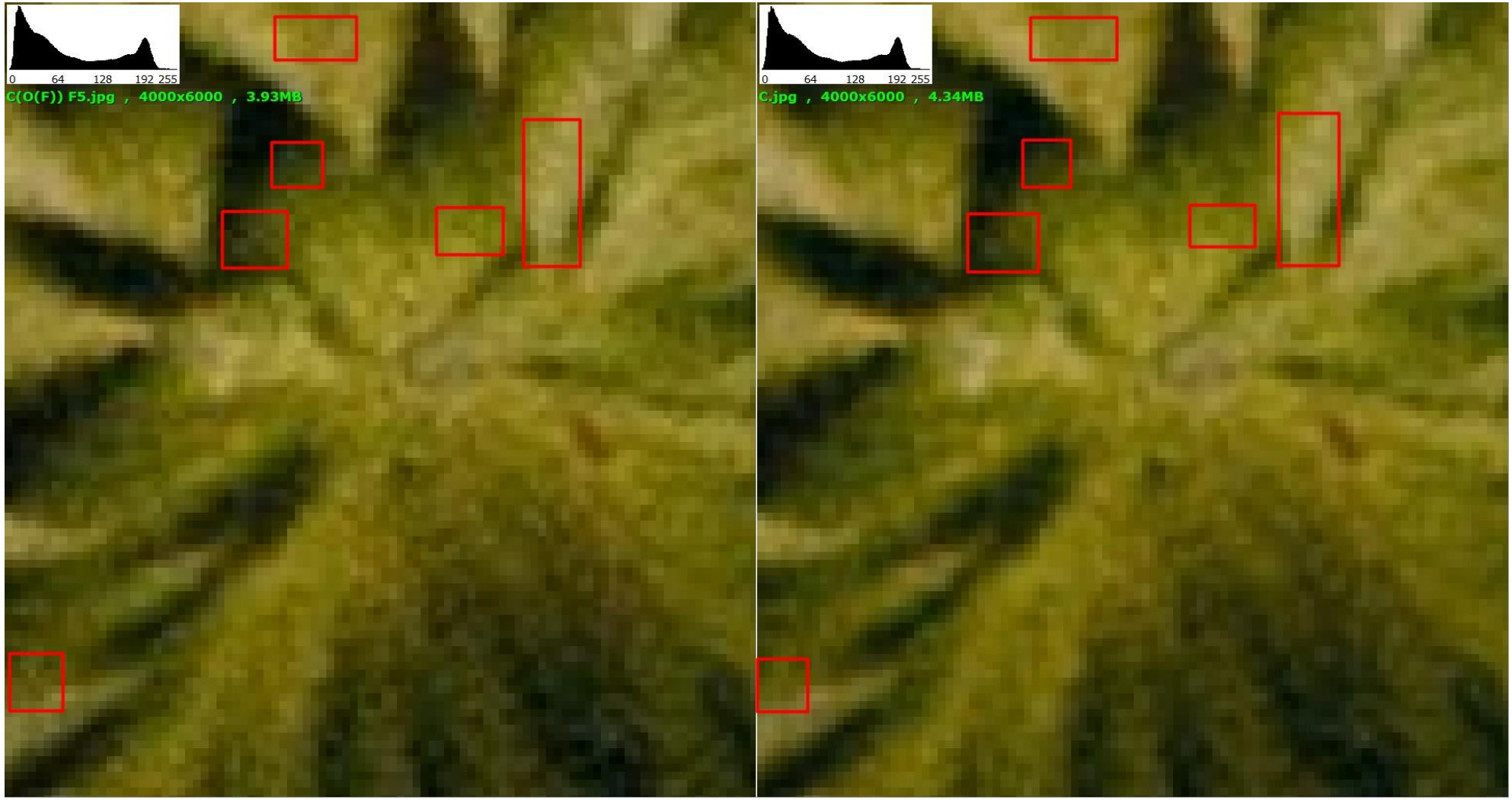
59 pav. O(F) F5.jpg ir O.jpg skirtumai

O(F) F5A 1/32.jpg ir O.jpg : (1000%)



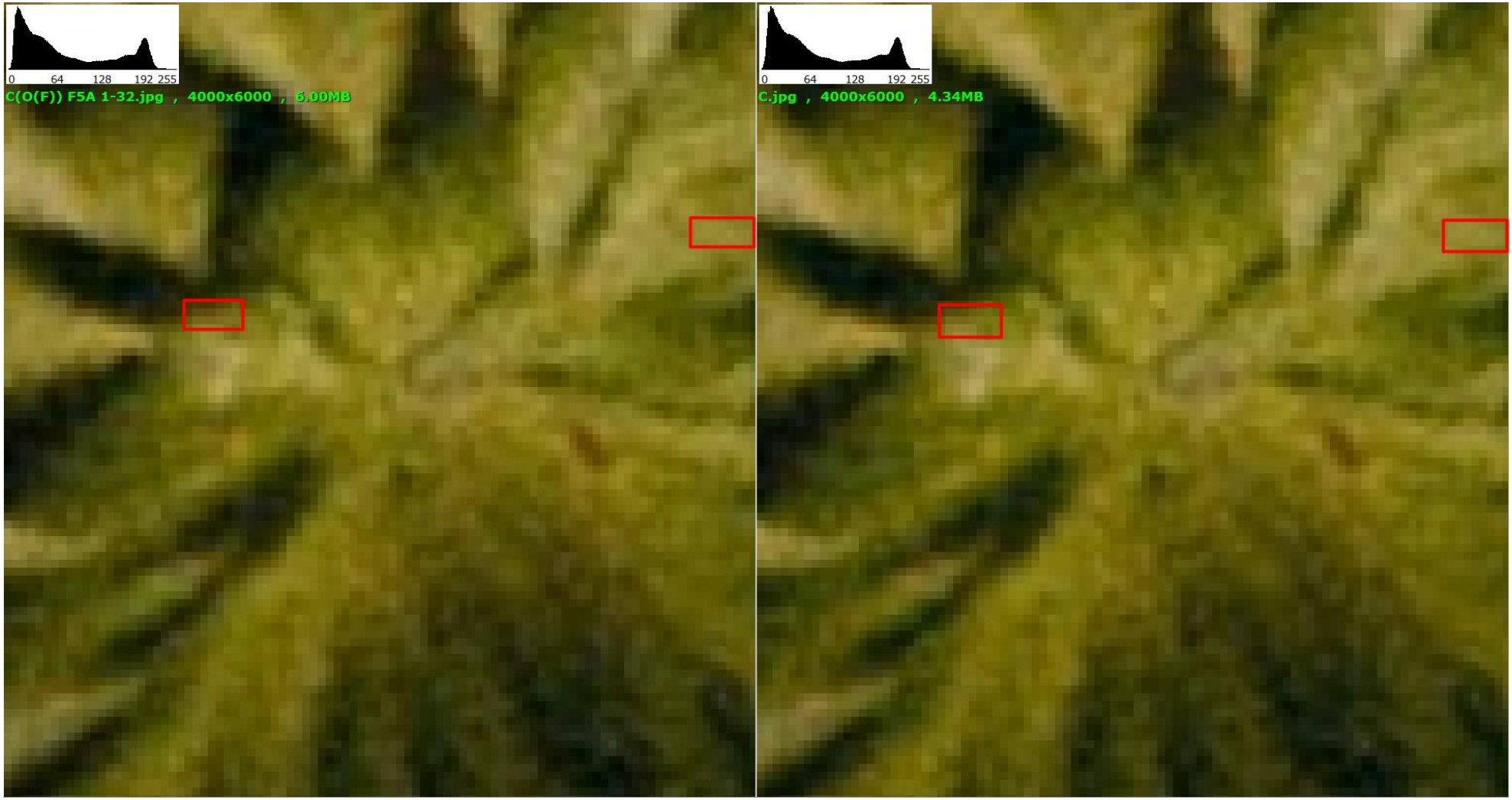
60 pav. O(F) F5A 1/32.jpg ir O.jpg skirtumai

C(O(F)) F5.jpg ir C.jpg : (1000%)



61 pav. C(O(F)) F5.jpg ir C.jpg skirtumai

C(O(F)) F5A 1/32.jpg ir C.jpg : (1000%)



62 pav. C(O(F)) F5A 1/32.jpg ir C.jpg skirtumai

C.jpg ir C(O(F)) F5A 16.jpg : (1000%) (didžiausia kompresijos santykio reikšmė 16 – prasčiausia įmanoma paveikslėlio kokybė)

63 pav. C.jpg ir C(O(F)) F5A 16.jpg skirtumai

Paveikslėliuose vaizduojami palyginimai tarp originalių paveikslėlių bei jų steganogramų sukurtų naudojant F5 bei F5A metodus. Iš gautų rezultatų buvo pastebėta, jog prie mažesnio kompresijos santykio, pastebimai pagerėja steganogramos kokybė, palyginus su F5 algoritmu.

Didesniame paveikslėlyje C(O(F)) skirtumai plika akimi beveik nepastebimi abiejuose metoduose, tačiau O(F) paveikslėlyje skirtumai pakankamai aiškūs, matomi raudonai apibrauktose vietose. Pateiktose nuotraukose matosi ir histogramos, sugeneruotos programinės įrangos, matome, jog O(F) paveikslėlyje F5 ir F5A metoduose histogramos pastebimai skiriasi, F5A metodo histograma geriau atitinka originalaus paveikslėlio histogramą, nėra didelių reikšmių šuolių, kokios matosi naudojant F5 metodą.

Taip pat eksperimentui paskutinėje nuotraukoje vaizduojamas F5A metodo rezultatas, su paveikslėlio kokybę pastebimai pakeitusiu 16 kompresijos santykiu, kuris rodo, kokia gali būti prasčiausia išgaunama kokybė naudojant F5A metodą.

4.6. Tyrimo išvados

Atlikus tyrimus buvo pastebėta, jog naujai sukurtas steganografinis metodas F5A naudojant skirtingus kompresijos santykius gali pastebimai pagerinti skirtingus steganogramos parametrus, lyginant su F5 metodu.

Nustatyta, jog F5A metodas mažesnio paveikslėlio O(F) atveju 1.716 karto pagerina MSE rodiklį, 1.121 karto pagerina PSNR rodiklį bei padidina paveikslėlio steganografinę talpą 1.693 karto. Didesnio paveikslėlio atveju pastebima net 3.156 karto padidėjusi MSE reikšmė, 1.121 karto pagerėjęs PSNR rodiklis bei 1.539 karto padidėjusi algoritmo talpa. Taip pat šalia pagerėjusių rodiklių pastebimas ir steganogramos failo dydžio pokytis. O(F) atveju 1.577 bei C(O(F)) atveju 1.544 karto padidėję steganogramos failo dydžiai.

Steganografinių metodų tvirtumo tyrimo metu, tarp F5 ir F5A metodų pakitimų nepastebėta, jų rezultatų atkurti nepavyko, dėl pakartotinio JPEG formato suspaudimo naudojamo paveikslėlių redagavimo programose.

Vizualiai palyginus steganogramas gautas naudojant F5 ir F5A metodus taip pat matomas paveikslėlio kokybės pagerėjimas, kuris patvirtina teigiamus rezultatus gautus MSE ir PSNR rodiklių skaičiavimuose. Pastebėtos konkrečios paveikslėlio vietos kuriose matosi pasikeitimai. Ryškiausi pastebimi pasikeitimai pastebėti mažesnio paveikslėlio O(F) atveju F5 metode, nors tą pačią vietą lyginant F5A metodo steganogramoje rastas vos vienas ryškiau pastebimas skirtumas.

IŠVADOS

Šio darbo metu buvo apžvelgtos skaitmeninės medijos saugos problemos bei jų sprendimo būdai ir pastebėta, jog esami skaitmeninės medijos apsaugos būdai dažniausiai suteikia pagrindą žalos atlyginimui, tačiau neužkerta kelio autorinių teisių pažeidimams. Pasinaudojant analizės dalyje gautais rezultatais buvo nustatytas poreikis naujam autorinių teisių apsaugos sprendimui ir išsikelti reikalavimai : Atpažinti neteisėtai naudojamą turinį, ir pranešti apie tai autorinių teisių savininkui bei apsaugoti atpažintą turinį nuo neteisėto jo panaudojimo.

Autorinių teisių apsaugos sprendimui įgyvendinti pagal išsikeltus reikalavimus buvo nuspręsta kurti naują steganografinį metodą „F5A“, kuris remiasi „F5“ algoritmu, papildomai integruojant optimizuotų kvantavimo lentelių naudojimą siekiant praplėsti steganografinio metodo talpą.

„F5A“ algoritmas buvo projektuojamas naudoti naujame „StegoGuard“ įskiepyje, kurio pagrindinė funkcija – nustatyti ar paveikslėlis nėra vogtas. Tai atlikti galima, specialiai užkoduotą steganogramą bandant atkoduoti du kartus – vieną su visiems žinomu raktu, kitą – su internetinio puslapio pavadinimu. Realizacijai buvo sukurta „StegoGuard“ įskiepio koncepcija ir ištestuotas jos veikimas.

Atlikus testavimą, buvo pastebėta, jog naujai sukurtas steganografinis metodas F5A naudojant skirtingus kompresijos santykius gali pastebimai pagerinti skirtingus steganogramos parametrus, lyginant su F5 metodu.

Nustatyta, jog F5A metodas mažesnio paveikslėlio O(F) atveju 1.716 karto pagerina MSE rodiklį, 1.121 karto pagerina PSNR rodiklį bei padidina paveikslėlio steganografinę talpą 1.693 karto. Didesnio paveikslėlio atveju pastebima net 3.156 karto padidėjusi MSE reikšmė, 1.121 karto pagerėjęs PSNR rodiklis bei 1.539 karto padidėjusi algoritmo talpa. Taip pat šalia pagerėjusių rodiklių pastebimas ir steganogramos failo dydžio pokytis. O(F) atveju 1.577 bei C(O(F)) atveju 1.544 karto padidėję steganogramos failo dydžiai.

Steganografinių metodų tvirtumo tyrimo metu, tarp F5 ir F5A metodų pakitimų nepastebėta, jų rezultatų atkurti nepavyko, dėl pakartotinio JPEG formato suspaudimo naudojamo paveikslėlių redagavimo programose.

Vizualiai palyginus steganogramas gautas naudojant F5 ir F5A metodus taip pat matomas paveikslėlio kokybės pagerėjimas, kuris patvirtina teigiamus rezultatus gautus MSE ir PSNR rodiklių skaičiavimuose. Pastebėtos konkrečios paveikslėlio vietos kuriose matosi pasikeitimai. Ryškiausi pastebimi pasikeitimai pastebėti mažesnio paveikslėlio O(F) atveju F5 metode, nors tą pačią vietą lyginant F5A metodo steganogramoje rastas vos vienas ryškiau pastebimas skirtumas.

Sukurta nauja autorinių teisių apsaugos koncepcija „StegoGuard“, naudojanti „F5A“ algoritmą pademonstravo gebėjimą atskirti vogtą turinį, nuo originalaus, taip užkertant kelią neteisėtai pasinaudoti vogtu turiniu.

Literatūros sąrašas

[1] Neha Navneet Patil, “INTELLECTUAL PROPERTY RIGHTS: CHALLENGES OF ENFORCEMENT OF PROTECTION OF COPYRIGHT LAWS IN THE DIGITAL ERA,” *Intellect. Prop. RIGHTS*, vol. 3, no. 4, p. 12, Aug. 2020.

[2] A. V. Venugopal, “Copyright concerns of digital images in social media,” *J. World Intellect. Prop.*, vol. 23, no. 3–4, pp. 579–597, Jul. 2020, doi: 10.1111/jwip.12147.

[3] P. Rai, “Copyright Laws and Digital Piracy in Music Industries : The Relevance of Traditional Copyright Laws in the Digital Age and How Music Industries should cope with the ongoing Piracy Culture,” *80*, 2020, Accessed: Dec. 22, 2021. [Online]. Available: https://uia.brage.unit.no/uia-xmlui/handle/11250/2727076

[4] A. Wallace, “Copyright,” 2020.

[5] Y. Chen, X. Hu, and F. Xiao, “Digital Media Copyright Protection Technology in the Age of All Media,” in *Data Processing Techniques and Applications for Cyber-Physical Systems (DPTA 2019)*, Springer, 2020, pp. 843–850.

[6] D. Megías, M. Kuribayashi, and A. Qureshi, “Survey on Decentralized Fingerprinting Solutions: Copyright Protection through Piracy Tracing,” *Computers*, vol. 9, no. 2, p. 26, 2020.

[7] F. Frattolillo, “Digital copyright protection: Focus on some relevant solutions,” *Int. J. Commun. Netw. Inf. Secur.*, vol. 9, no. 2, p. 282, 2017.

[8] “13 Image Protection Services for Professional Photographers -,” *PDN Online*, Apr. 04, 2019. https://pdnonline.com/photography-business/copyright-law/image-protection-services-for-professional-photographers/ (accessed Dec. 22, 2021).

[9] “11 Free Copyright Tools for Photographers and Artists,” *Plagiarism Today*, Jun. 17, 2010. https://www.plagiarismtoday.com/2010/06/17/11-free-copyright-tools-for-photographers-and-artists/ (accessed Dec. 22, 2021).

[10] “The Ancient Practice of Steganography: What Is It, How Is It Used and Why Do Cybersecurity Pros Need to Understand It,” *Default*. https://www.comptia.org/blog/what-is-steganography (accessed Dec. 22, 2021).

[11] M. Nosrati, R. Karimi, and M. Hariri, “An introduction to steganography methods,” *World Appl. Program.*, vol. 1, pp. 191–195, Aug. 2011.

[12] K. S. Mohamed, “Data Hiding: Steganography and Watermarking,” in *New Frontiers in Cryptography*, Springer, 2020, pp. 89–98.

[13] L. K. Saini and V. Shrivastava, “A survey of digital watermarking techniques and its applications,” *ArXiv Prepr. ArXiv14074735*, 2014.

[14] P. Kadian, S. M. Arora, and N. Arora, “Robust digital watermarking techniques for copyright protection of digital data: A survey,” *Wirel. Pers. Commun.*, pp. 1–25, 2021.

[15] A. Westfeld „F5—A Steganographic Algorithm High Capacity Despite Better Steganalysis“, 2001.

[16] C.C. Chang, T.S. Chen, L.Z. Chung, A steganographic method based upon JPEG and quantization table modification. Inf. Sci. 141, 123–138 (2002).

[17] D.M. Monro, B.G. Sherlock, Optimal quantization strategy for DCT image compression. IEE Proc., Vis. Image Signal Process. 143(1), 10–14 (1996).

5. PRIEDAI

5.1. Optimizuotos kvantavimo lentelės gautos prie skirtingų kompresijos santykių

Naudojama standartinė kvantavimo lentelė F5 algoritmui:

```
16 11 10 16 24 40 51 61
12 12 14 19 26 58 60 55
14 13 16 24 40 57 69 56
14 17 22 29 51 87 80 62
18 22 37 56 68 109 103 77
24 35 55 64 81 104 113 92
49 64 78 87 103 121 120 101
72 92 95 98 112 100 103 99
```

Sugeneruotos optimizuotos kvantavimo lentelės, pagal skirtingą kompresijos santykį F5A algoritme:

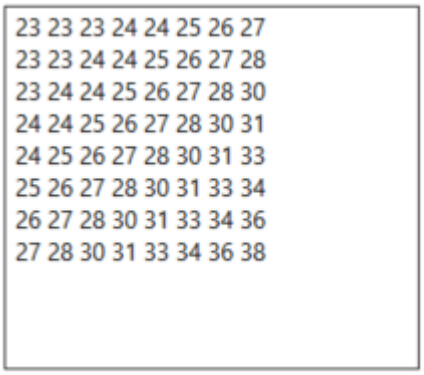
Kompresijos santykis 128

```
281 396 692 1144 1742 2478 3349
4349
396 692 1144 1742 2478 3349 4349
5475
692 1144 1742 2478 3349 4349 5475
6725
1144 1742 2478 3349 4349 5475
6725 8095
1742 2478 3349 4349 5475 6725
8095 9584
2478 3349 4349 5475 6725 8095
```

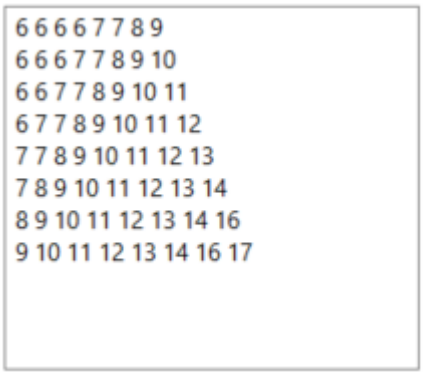
Kompresijos santykis 32 :

```
74 80 95 119 150 188 232 284
80 95 119 150 188 232 284 342
95 119 150 188 232 284 342 406
119 150 188 232 284 342 406 477
150 188 232 284 342 406 477 554
188 232 284 342 406 477 554 637
232 284 342 406 477 554 637 725
284 342 406 477 554 637 725 820
```

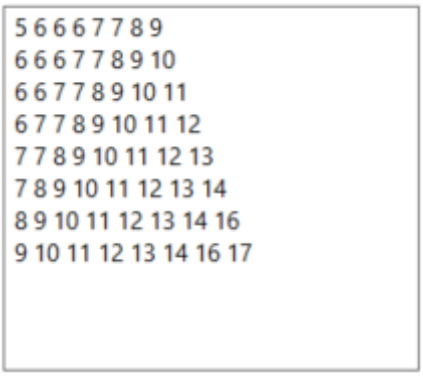
Kompresijos santykis 8 :



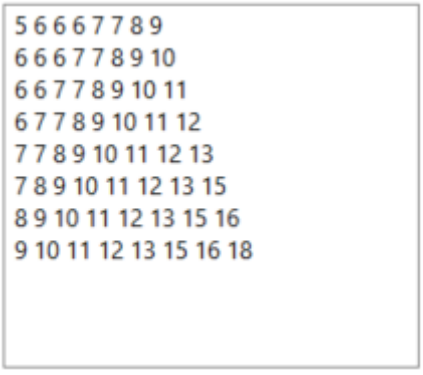
Kompresijos santykis 1/8 :



Kompresijos santykis 1/32 :



Kompresijos santykis 1/128 :



Kompresijos santykis 0 :

5 6 6 6 7 7 8 9
6 6 6 7 7 8 9 10
6 6 7 7 8 9 10 11
6 7 7 8 9 10 11 12
7 7 8 9 10 11 12 13
7 8 9 10 11 12 13 15
8 9 10 11 12 13 15 16
9 10 11 12 13 15 16 18