Digitalna forenzika

Steganografija slika – sakrivanje podataka metodom diskretne kosinusne transformacije

Mentor:
prof. dr Bratislav Predić

Student:
Marija Milošević 1045

Sadržaj

1. Uvod 3
2. Steganografija 4
2.1. Elementi procesa steganografije 4
2.2. Faktori koje treba razmotriti 6
2.3. Klasifikacije steganografskih metoda 7
3. Steganografija slika 8
3.1. Tehnike steganografije slika 10
3.1.1. LSB 12
3.1.2. DCT 13
3.1.3. DFT 14
3.2.4. DWT, IWT 15
3.2. Postojeći softver 16
3.3. JPEG 18
3.4. Predložena rešenja 23
3.5. Enkripcija 25
3.6. Opis implementacije 26
4. Zaključak 34
Literatura 35

1. Uvod

Internet i bežične mreže omogućavaju razmenu multimedijalnih informacija. Različiti softveri i uređaji su omogućili korisnicima širom sveta da pristupe, razvijaju i modifikuju multimedijalne objekte. U skladu s tim, vlade i organizacije se trude da sačuvaju privatnost svojih podataka i tehnika poslovanja. Jedna od metoda koja se koristi je steganografija. Steganografija je tehnika skrivanja tajnih podataka unutar običnih, dostupnih fajlova ili poruka, kako bi se izbegla njihova detekcija. Tajni podaci se izvlače na dolaznom kraju konekcije, s tim što će samo određeni primaoci moći da dekodiraju skrivene informacije. Steganografija se može kombinovati s enkripcijom koja bi bila dodatni korak sakrivanja i zaštite podataka. Različiti motivi se uočavaju za upotrebu steganografije, pa se tako i kreiraju različite metode, koje pritom postaju sve naprednije. Uvek će postojati pojedinci ili grupe koje pokušavaju dekriptovanje informacije i nalaženje nečeg skrivenog.

Trenutni rast interesovanja za steganografiju se pripisuje povećanom obimu upotrebe digitalnih medija i razvoju interneta. Takvi objekti postali su zgodno mesto za sakrivanje tajnih informacija. U poređenju s drugim digitalnim objektima, polje steganografije slika je trenutno najrazvijenije, sa različitim metodama za najčešće formate slika. Ove metode iskorišćavaju slabosti ljudskog sistema za vid i njegove osetljivosti na određene pojave.

Cilj ovog rada je predstavljanje osnovnih koncepata steganografije, uz poseban osvrt na steganografiju slika. Fokus će biti na jednoj od metoda koje informacije sakrivaju u frekventnom domenu – diskretnoj kosinusnoj transformaciji (DCT). Kako digitalne slike dolaze u različitim formatima, ovde je predstavljeno sakrivanje po uzoru na JPEG kompresiju, kao jednom od najčešće korišćenih formata. Biće dat opis predloženih tehnika postojećih radova, kao i opis algoritama korišćenih za enkripciju i kompresiju sakrivenih informacija.

2. Steganografija

Steganografija je termin koji se upotrebljava za sakrivanje fajla, poruke, slike ili videa unutar drugog fajla, poruke, slike ili videa. Prvi put je zabeležena upotreba ovog termina 1499. godine, a smatra se da su je prvi koristili Grci još 440. god. p.n.e., ostavljajući poruke na glavama sluga. U novijem dobu, popularno je bilo nevidljivo mastilo, posebno tokom drugog svetskog rata za pisanje tajnih poruka. Zatim su Nemci uveli tehniku mikrotačaka gde su se one smatrale za fotografije, koje su uključivane u pismo ili kovertu, a koje su zbog svoje veličine bile neprimetne.

I sami dokumenti su korišćeni za sakrivanje poruka; tekst se unutar dokumenta može sakriti preko „nula cifara“ (eng. *null cipher*), tako kamuflirajući poruku u normalan tekst. Na primer, u Drugom svetskom ratu, ovako je izgledala poruka poslata od strane nemačkog špijuna:

“Apparently neutral’s protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-products, ejecting suets and vegetable oils”.

Izvlačenjem svakog drugog slova svake reči, otkriva se skrivena poruka:

“Pershing sails from NY June 1.“

Steganografija se primenjuje za različite stvari, od kojih su interesatniji autentifikacija fajlova, anotacija, bankarske transakcije i poboljšane strukture podataka. Vojska i više vladinih agencija istražuju polje steganografije za svoje tajne transmisije informacija. Takođe žele da presretnu tajne informacije kriminalaca i terorista. Istraživanja su pokazala da je steganografija najčešće korišćena u napadu Al Kaide na Svetski trgovinski centar. Pored toga, napadači mogu slati Trojance i viruse, s ciljem kompromitovanja osetljivih sistema.

2.1. Elementi procesa steganografije

Proces steganografije se sastoji od 3 osnovne komponente: nosioca poruke, tajne poruke i tajnog ključa. Nosilac može biti slika, video, MP3, čak i TCP/IP paket. To je prenosni medijum za skrivenu poruku. Ključ se koristi za razotkrivanje tajne poruke. To može biti šifra, obrazac, čak i video, i on može rasuti poruku širom nosioca.

Koncept steganografije podrazumeva da postoje dva korisnika, A i B, koja žele da tajno komuniciraju. Međutim, svu komunikaciju ispituje C preko lokalnog servera ili rutera. U opštem steganografskom modelu, A želi da pošalje tajnu poruku m korisniku B. Da bi to uradio, A je ugrađuje u objekta nosioca c, i dobija *stegoobjekat* s. Taj stegoobjekat se zatim šalje preko nekog javnog kanala. Cilj steganografa je zapravo proizvesti stegoobjekat s od nosioca c i tajne poruke

m, koji će tajnu poruku preneti. Po standardnoj definiciji steganografije, tehnika za ugrađivanje poruke je nepoznata osobi C, a deljena između A i B. Međutim, uopšteno se smatra da korišćeni algoritam nije tajan, već samo ključ koji se koristi u algoritmu, a koji treba čuvati tajnim između dve uključene strane – što se poklapa s Kirhofovim principom na polju kriptografije. On tvrdi da se bezbednost kriptografskog sistema treba bazirati isključivo na ključu. Kako bi steganografija ostala nedetektovana, nemodifikovani nosilac mora se čuvati u tajnosti, jer ukoliko se otkrije, poređenje između njega i stegoobjekta će odmah otkriti promene.

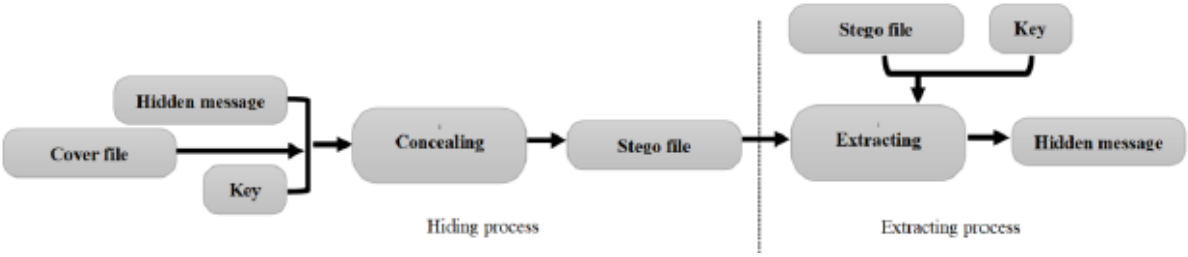
Steganografska šema se matematički može definisati na sledeći način:

Neka je *M* skup poruka za ugrađivanje, a *C* skup nosilaca poruke. Neka je *k* stego ključ iz skupa stego ključeva *K*. Dva mapiranja su uključena u steganografsku šemu, ugrađivanje (*eng. Embedding*) i ekstrakcija (*eng. Extraction*):

Emb: CxKxM → C'

Ext: C' → M, pri čemu važi $Ext(Emb(c,k,m)) = m, \ \forall c \in C, \forall k \in K, \forall m \in M.$

C' = *Emb*(c,k,m) predstavlja generisani stegoobjekat. Na slici 1 prikazan je celokupan proces steganografije: ugradnja poruke, generisanje stegoobjekta, njegovo slanje, a zatim i ekstrakcija poruke na drugoj strani komunikacionog kanala.



Slika 1. Proces steganografije [4]

Modifikacija objekta nosioca menja njegova statistička svojstva, tako da se distorzije u svojstvima rezultujućeg stegoobjekta mogu detektovati. Grana koja se bavi nalaženjem ovakvih distorzija je statistička stegoanaliza.

Steganografija se razlikuje od kriptografije, ali njihova kombinacija može poboljšati bezbednost zaštićenih informacija i sprečiti detekciju tajne komunikacije. Dok se kriptografija bavi zaštitom sadržaja same poruke, steganografija na to dodaje i skrivanje činjenice da se tajna poruka zapravo tu nalazi i šalje. Prednost steganografije u odnosu na samu kriptografiju je to što tajna poruka ne privlači pažnju na sebe. Kod enkriptovanog fajla, poruka ili mrežni sadržaj jasno su označeni i vidljivi (iako je sadržaj zbog enkripcije nerazumljiv), dok steganografske tehnike pomažu u sakrivanju prisustva bilo kakve tajne. Nijedna od ove tehnike nije jedinstveno i najbolje rešenje za

probleme privatnosti u otvorenim sistemima. Bez obzira na to, svaka od njih može doprineti poboljšanju privatnosti u komunikacijama.

Tehnika menjanja medijuma neprimetno kako bi se dodale informacije o njemu poznata je kao *watermarking*. Steganografija i watermarking su konceptualno različite tehnike – kod watermarkinga komunikaciju predstavlja nosilac, a zaštita je u sakrivenim podacima, dok su kod steganografije komunikacija tajni podaci, a nosilac je samo neka vrsta omotača za te podatke. Za razliku od watermarkinga, steganografija nije primarno namenjena da spreči uklanjanje ili menjanje skrivene poruke, već naglasak baca na to da se poruka ne može detektovati.

2.2. Faktori koji se razmatraju

Različite karakteristike definišu performanse određene steganografske metode. Efikasnost steganografske tehnike se može odrediti upoređivanjem stegoobjekta s objektom nosiocem, pri čemu se posmatraju sledeći faktori:

- Neprimetnost – glavni zahtev steganografije; ovo podrazumeva da ne postoji metod koji može odrediti postoji li skrivena poruka u slici. Razlikujemo perceptivnu i statističku (algoritamsku) neprimetnost. Perceptivna neprimetnost odnosi se na meru u kojoj skrivene informacije moraju ostati neopažene ljudskim čulima – sluhu, u slučaju audio steganografije ili vidu, u slučaju steganografije slika. Statistička neprimetnost podrazumeva neprimetnost sakrivenih informacija u odnosu na statističku ili algoritamsku analizu. Treba napomenuti da cilj steganografije neće biti dostignut ukoliko je išta od skrivenih informacija na bilo koji način makar i naslućeno, ako ne potpuno primetno.
- Kapacitet ugradnje – odnosi se na maksimalnu dužinu tajne poruke koja se može ugraditi u objekat nosilac bez izazivanja vizuelne ili statističke detekcije. Uopšteno, steganografske tehnike se razvijaju tako da maksimalno iskorišćavaju kapacitet ugradnje objekta nosioca. Postići visok kapacitet je kod steganografije jako bitno jer je glavni cilj upravo transmisija informacija. Digitalne slike su zgodne za primenu u steganografiji jer imaju visok nivo redundantnosti pri prikazivanju i primeni u svakodnevnom životu.
- Robusnost – otpornost na manipulaciju nosioca objekta. Reč je o sposobnosti ugrađivanja podataka tako da oni ostanu neizmenjeni ukoliko se stegoobjekat podvrgne nekim od osnovnih transformacija – kod stegoslike, to može biti filtriranje, izoštravanje, rotacija, itd. Ovo je jako poželjna karakteristika kod steganografskih tehnika. Robusnost se odnosi na transformacije koje sprovode lica koja ne sumnjaju na postojanje skrivenih informacija. Robusnost kod namernih napada predstavljaće bezbednost steganografske metode. Ako pretpostavimo da napadač veruje da neki medijum krije informacije, ali nema stego ključ na raspolaganju, on se u to može uveriti samo ako informacija izvuče namernim napadom. Zato, ako je steganografska šema bezbedna i on ne uspe da povrati informacije, nikada ne

može biti 100% siguran da li je reč o nekoj anomaliji u podacima ili zaista skrivenim informacijama.

- Kompleksnost i troškovi – zahtev koji nije obavezan, ali je poželjno razmotriti ga. Kod nekih primena, kompleksnost računice je prilično bitna, na primer, kod aplikacija koje strimuju muziku uživo i kriju informacije u isto vreme. Tada je cilj smanjiti kompleksnost i troškove. Najčešće se kod steganografije koristi transmisija informacija u realnom vremenu.

Idealna steganografska metodologija trebalo bi da ima visok nivo prve tri navedene karakteristike. Nažalost, nijedna metoda ne može zadovoljiti sve stavke. U većini slučajeva, postojaće kompromisi u zavisnosti od zahteva specifične primene. Najčešće je cilj dobiti visoku neprimetnost i kapacitet, smanjujući time bitnost robusnosti. Međutim, ako ostavimo robusnost po strani i posmatramo druga dva faktora – povećanjem neprimetnosti, pojačavamo glavni cilj steganografije, ali količina prenetih informacija možda neće biti dovoljna. Ukoliko se određena metoda ne fokusira na robusnost, može doći do toga da skrivene informacije budu krhke, te ih i trivijalne modifikacije nad stegoobjektom mogu uništiti.

Steganografski sistemi za JPEG format deluju interesantno jer mogu funkcionisati u prostoru transformacija, te na njih ne utiču vizuelni napadi. O ovome će više biti reči u poglavlju 3.

2.3. Klasifikacije steganografskih metoda

Steganografske metode se mogu različito klasifikovati, i u literaturi se nailazi na par znatno drugačijih podela. Steganografija se široko može podeliti na tri tipa – tehničku, lingvističku i digitalnu. Tehnička steganografija primenjuje naučne metode za skrivanje tajnih poruka, dok lingvistička koristi pisani prirodni jezik. Nevidljivo mastilo je primer tehničke, a null cipher i semagrami su primer lingvističke steganografije. Digitalna steganografija, razvijena s nastankom računara, koristi računarske fajlove i digitalne podatke za sakrivanje.

Klasifikacija po tipu objekta nosioca – Kada govorimo o različitim nosiocima, steganografiju možemo podeliti na steganografiju teksta, slika, audio ili video steganografiju, steganografiju animacija, itd. Tekstualna steganografija ugrađuje informacije promenom nekih od osnovnih karakteristika teksta, što uvodi neke promene u dokumentu, ali ljudsko oko njih ne zapaža. Koriste se tehnike koje utiču na broj tabova, bele prostore, velika slova, itd. Multimedijalni fajlovi su idealni za steganografsku transmisiju zbog svoje veličine. Na primer, pošiljalac može početi bezazlenom slikom i podesiti boju svakog hiljaditog piksela da odgovara slovu azbuke. Promena je toliko suptilna da je onaj koji je eksplicitno ne traži ni ne primećuje. Steganografija slika koristi intenzitete piksela za sakrivanje informacija, pri čemu je većim slikama nekad neophodna kompresija za izbegavanje detekcije. Tehnike audio steganografije ugrađuju informacije u WAVE,

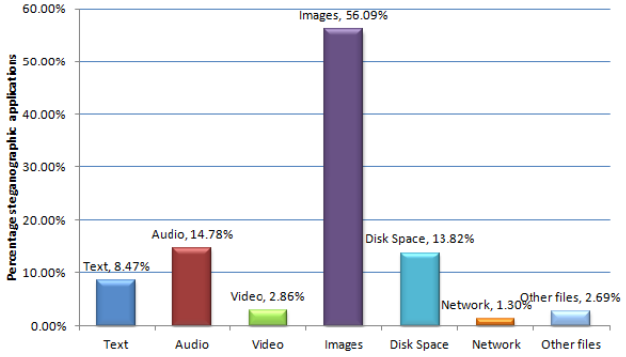
MIDI, MPEG formate za postizanje copyright-a, ali i skrivene komunikacije. Kod video steganografije, informacije se ugrađuju u nosioca signala tako da se iskoriste ograničenja osetljivosti ljudskog oka. Još jedna opcija je mrežna steganografija gde se za nosioca uzima neki od mrežnih protokola. U OSI modelu postoje kanali u kojima se informacije mogu kriti u neiskorišćenim bitovima zaglavlja TCP/IP polja.

Klasifikacija po domenima ugradnje – ovde postoje metode zasnovane na originalnom domenu, na domenu transformacije i na domenu kompresije. Metode zasnovane na originalnoom domenu se odnose na klasu metoda koje direktno modifikuju originalne podatke cover objekta po nekim pravilima, npr. zamenom redundantnog dela nosioca tajnim informacijama. Algoritmi zasnovani na domenu transformacije prvo izvršavaju neku transformaciju na originalnom nosiocu (poput FFT-a, DCT-a, DWT-a) da bi se dobili koeficijenti transformacija, a zatim njihovom modifikacijom ugrađuju informacije. S razvojem tehnologija za kompresiju, mnogo objekata nosilaca se obično smešta u komprimovanom formatu, pa se pažnja polako preusmerava na ugrađivanje informacija direktno u komprimovanom domenu. Tehnike u komprimovanom domenu kombinuju skrivanje informacija i proces kompresije, što može efikasno sprečiti napade koji dolaze od perceptivnog kodiranja. Na primer, u toku procesa kompresije JPEG slike, mogu se ugraditi tajne informacije modifikacijom koeficijenata transformacije i faktora kvantizacije. Ovo će detaljnije biti objašnjeno u narednom poglavlju.

3. Steganografija slika

Najčešće korišćene steganografske metode su ugrađivanje tajnih informacija u digitalne slike, kao što je već pomenuto u poglavlju 2. Bilo koji tekst ili digitalna slika može se ugraditi u digitalnu sliku. Slika 2 prikazuje broj aplikacija koje kriju informacije u elektronskim medijima od 2008. godine. Vidi se da je više od polovine primena upravo koristilo slike kao nosioce, iza čega sledi audio steganografija kao druga razvijena grana.

Cover Media	Count	Percentage
Text	255	8.47%
Audio	445	14.78%
Video	86	2.86%
Images	1689	56.09%
Disk Space	416	13.82%
Network	39	1.30%
Other files	81	2.69%



Slika 2. Učestalost različitih tipova nosilaca poruke od 2008. godine.

Digitalna slika je predstavljena matricom numeričkih vrednosti koje predstavljaju intenzitete za različite tačke koje se zovu pikseli. Pikseli čine rasterske podatke slike. Dubina bita se odnosi na količinu bitova u šemi boja i to je količina bitova koja se koristi za svaki piksel. Najmanja dubina bita u trenutnim šemama je 8, tj. 8 bitova se koristi za opis boje svakog piksela. Monohromantske i grayscale slike koriste 8 bitova za svaki piksel i imaju opciju predstavljanja ukupno 256 različitih boja ili nijansi sive, dok se slike digitalnih boja primarno čuvaju u 24-bitnim fajlovima, gde se koristi RGB model boja. Tako u svakom pikselu postoji po 256 različitih nijansi crvene, zelene i plave, što donosi preko 16 miliona kombinacija.

Kako bi se slika prikazala u razumnom vremenu, moraju se primeniti metode za redukciju veličine fajla - kompresija. Postoje dve vrste kompresije – s gubitkom (*eng.* lossy) i bez gubitaka (*eng. lossless*), koje se razlikuju u implementaciji. Kompresija s gubicima dovodi do redukcije veličine fajla tako što se rešava dodatnih podataka originalne slike. Ona uklanja detalje koji su previše sitni da bi ih ljudsko oko detektovalo, što dovodi do aproksimacije originalne slike, iako to nije najtačnija replika. Format slike koji koristi ovu metodu kompresije je JPEG. Tehnike kompresije igraju bitnu ulogu u odabiru steganografskog algoritma: lossy kompresija povećava šanse gubitka dela tajne poruke jer će višak podataka slike biti eliminisan. S druge strane, kompresija bez gubitaka sprečava gubitak i najmanjeg dela slike, ali ne može kompresovati fajl u manju veličinu. Za oba tipa kompresije, predloženi su različiti steganografski algoritmi.

Pri ugrađivanju tajnih informacija u sliku, pikseli slike se menjaju u skladu s informacijama koje se ugrađuju. Većina digitalnih formata je pogodna za steganografiju, ali se ipak oni sa većim nivoom redundantnosti ili šuma smatraju prikladnijim. Slike, audio i video fajlovi posebno se pridržavaju ovog zahteva. Najpopularniji formati slika na internetu su GIF, JPEG i, u manjoj meri, PNG. Većina razvijenih tehnika koriste neki od ovih formata, uz par primera iz literature koji koriste BMP format zbog jednostavne strukture podataka.

U steganografiji, odabir slike nosioca je kritičan jer utiče na dizajn steganografskog sistema i neophodne sigurnosti. Bitno je ne koristiti nosioce s velikim blok-područjima jer je promene tu lakše detektovati. Slike sa malim brojem boja takođe ne treba birati. Popularne slike poput Mona Lize se ne trebaju uopšte koristiti, pa je za ljude koji se bave steganografijom bolje kreirati sopstvene slike nosioce. Zbog zahteva steganografije da se tajna ne sme detektovati, slika nosilac treba se kriti, jer bi se u suprotnom mogle otkriti promene pri poređenju nje sa stegoslikom. Neki čak preporučuju i da se nakon upotrebe nosilac uništi ili ne koristi ponovo.

Koncept neprimetnosti, koji je glavna stavka kod steganografije, može se proceniti različitim metrikama koje nisu pomenute u poglavlju 2.2. Najučestalije su subjektivni testovi i PSNR (*eng. Peak-Signal-to-Noise-Ratio*). Neprimetnost steganografske metode je visoka ako ona kreira sliku koja nakon unošenja podataka ostaje skoro bez distorzija i ljudsko oko ne može lako detektovati razliku. Subjektivne testove sprovode ljudi koji traže vizuelne razlike između slika, pokušavajući da odrede koja od dve slike je originalna slika nosilac. Ukoliko je procenat uspeha iznad 50%, zaključuje se da je poruka nevidljiva. Pravila i preporuke za ove testove definiše ITU (*eng.*

International Telecommunication Union). Za razliku od njih, PSNR je tehnički pristup za procenu pravog kvaliteta stegoslike. Ovo je inženjerski termin za odnos maksimalne moguće jačine signala i jačine šuma koji utiče na vernost reprezentacije tog signala. PSNR se najčešće koristi da odredi kvalitet rekonstrukcije slike, poređenjem stegoslike s originalnom slikom. Što je veći PSNR, to je bolji kvalitet slike, tj. manja distorzija. Za računanje PSNR-a, biće neophodno izračunati MSE (*eng. Mean Square Error*), kojom se meri distorzija slike. Formula za MSE data je u nastavku, pri čemu je I slika nosilac, K stegoslika, a m*n je broj piksela.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|I(i,j) - K(i,j)\|^2$$

Imajući ovu formulu u vidu, PSNR se računa po sledećoj formuli, gde *MAXi* predstavlja maksimalnu vrednost piksela na slici (za 8-bitne grayscale slike, to će biti 255). Tipične vrednosti za PSNR u kompresiji slika i videa su između 30 i 50 dB.

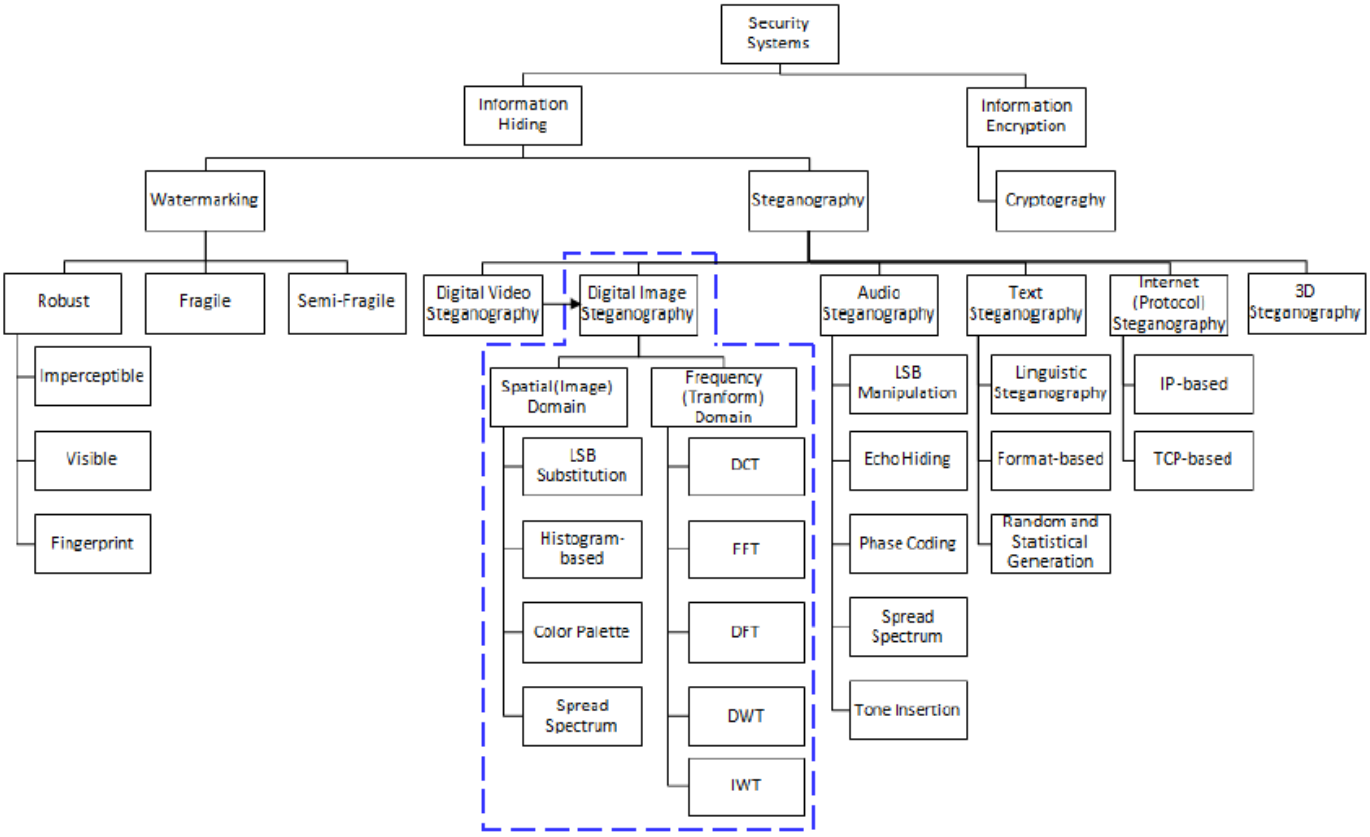
$$PSNR = 10.log_{10}\left(\frac{MAX_i^2}{MSE}\right) = 20.log_{10}\left(\frac{MAX_i}{\sqrt{MSE}}\right)$$

3.1. Tehnike steganografije slika

Kako su u poglavlju 2 pomenute osnovne klasifikacije metoda steganografije, ovde će detaljnije biti opisana podela steganografija slika u zavisnosti od domena. Na slici 3 data je podela steganografije, s naglaskom na različite tehnike steganografije slika.

Steganografija se jednostavno može postići nadovezivanjem tajne poruke tekstualnog fajla *message.txt* u JPEG fajl *cover.jpg* i dobiće se stegoslika *stego.jpg*. Poruka se pakuje i ubacuje nakon EOF taga. Kada se *stego.jpg* pregleda bilo kojom aplikacijom za editovanje slika, sve nakon EOF taga se ignoriše. Međutim, kada se otvori u Notepad-u, poruka se otkriva nakon prikazivanja nekih neurednih podataka. Ugrađena poruka neće narušiti kvalitet slike. Nažalost, ova tehnika neće se odupreti nijednoj vrsti editovanja stegoslike ili napadima stegoanalitičara.

Još jedna naivna implementacija je nadogradnja skrivenih podataka u dodatne informacije fajla slike, što je standard koji koriste proizvođači digitalnih kamera za skladištenje metapodataka o slici, a nalaze se u zaglavlju fajla. Ovo nije pouzdana metoda jer ima slične mane kao EOF metoda. Nije uvek preporučljivo kriti podatke direktno, bez enkripcije, kao u ovom primeru.



Slika 3. Steganografija slika – jedna od podela [1]

Spacijalni domen steganografije odnosi se na metode u kojima se skrivanje podataka vrši direktno na vrednostima piksela slike nosioca tako da efekat poruke nije vidljiv na njoj - bar ne ljudskom oku. Metode spacijalnog domena uključuju manipulaciju bitovima primenom LSB tehnike (*eng. Least Significant Bit*) uz manipulaciju šumom. Ove metode su široko korišćene i uglavnom je reč o jednostavnim sistemima. Formati slika koji se koriste u ovoj steganografiji su uglavnom lossless – bez gubitaka – gde metode jako zavise od formata slike. Uočeno je da su ove metode jednostavne za ugrađivanje informacija. Međutim, značajno su ranjive na male modifikacije. Jednostavno se mogu primeniti metode obrade signala za one koji žele da unište skrivene informacije. Od tehnika iz ovog domena koje su prikazane na slici 3, detaljnije će biti objašnjena LSB tehnika i njene varijacije.

Tehnike frekventnog domena koriste specifične karakteristike domena slike kako bi u nju unele podatke, prvobitno transformišući sliku u nekom od tih domena. Informacije se ugrađuju kroz modifikaciju određenih koeficijenata transformacije slike nosioca. Ovde se, dakle, podaci ugrađuju u transformisanu sliku umesto u piksele, pa se slika nakon toga ponovo vraća u spacijalni domen. Prednost ovih algoritama je što se sakrivene informacije šire na veliki broj piksela ili čak na celu sliku, što smanjuje mogućnosti njihovog uklanjanja. Većina ovih metoda ne zavisi od formata slike, gde ugrađena slika može biti podvrgnuta i lossy i lossless kompresiji. U poređenju sa spacijalnim

domenom, metode frekventnog domena su kompleksnije, ali je robusnost zadovoljena – čak i ako se ugradi dosta sadržaja, vizuelni efekat je i dalje jako dobar. Sada je već široko prihvaćeno da se informacije ugrađuju u koeficijente srednjih frekvencija domena transformacije slike nosioca. Do ovoga se došlo jer visoke frekvencije imaju malo efekta na kvalitet originalne slike, dok niske frekvencije mogu postići dobru robusnost, pa je deo srednjih frekvencija dobar kompromis između neprimetnosti i robusnosti. Od tehnika iz ovog domena koje su prikazane na slici 3, detaljnije će biti objašnjene DCT, DFT i DWT tehnike.

U nastavku je dat tabelarni uporedni prikaz karakteristika spacijalnog i frekventnog domena.

	Prednosti	Mane
Spacijalni domen	• Visok kapacitet ugradnje • Jednostavni sistemi • Održavanje oštrine slike	• Uglavnom zavisi od formata slike • Ugrađene informacije se lako mogu detektovati statističkom analizom • Loša robusnost na lossy kompresiju i šum • Loša robusnost za kropovanje ili rotiranje slike
Frekventni domen	• Tipično nezavisni od formata slike • Nema „vizuelnih napada“	• Više računanja, kompleksniji sistemi • Loša robusnost za statističke analize drugog reda

Pored ova dva domena, u razvoju je i adaptivna tehnika (adaptivna steganografija), kao specijalan slučaj prethodne dve metode. Ona uzima statističke globalne karakteristike slike pre pokušaja da modifikuje LSB/DCT koeficijente. Razvoj adaptivne tehnike zahteva puno znanje reprezentativnih osobina slike nosioca kako bi se odlučilo gde treba praviti promene.

3.1.1. LSB

LSB tehnika je najčešća i najlakša metoda za skrivanje poruka. Ovde se poruka krije u najmanje značajnim bitovima piksela slike. Menjanjem tih bitova piksela, ne uvodi se značajna razlika u sliku, pa stegoslika liči na originalnu sliku. Kod 24-bitnih slika, 3 bita piksela se mogu koristiti za LSB supstituciju jer svaki piksel ima različite komponente za crvenu, zelenu i plavu. Ugrađivanje tajne poruke u najmanje značajne bitove može se vršiti sekvencijalno ili se oni mogu birati

nasumično. Odabir zavisi od tajnog ključa koji se deli između pošiljaoca i primaoca. Za ugradnju neke veće tajne poruke, informacije se često kriju u drugom i trećem bitu svakog od piksela. Uopšteno, LSB algoritam ima veći kapacitet u poređenju s drugim metodama – velika količina informacija se može ugraditi.

LSB steganografija uključuje LSB zamenu i LSB poklapanje (*eng. LSBM – LSB Matching*). Kod zamene, LSB-ovi slike se direktno menjaju tajnim bitovima poruke. Međutim, ona dozvoljava laku detekciju stegoanalizom zbog svoje jednostavnosti i nebalansirane modifikacije piksela. LSBM je modifikovana verzija: ako se LSB piksela slike nosioca poklapa s tajnim bitom, vrednost se ne menja, u suprotnom, 1 se dodaje ili oduzima, po slučajnom odabiru. LSBM smanjuje verovatnoću detekcije poruke, ali dolazi s određenom distozijom. LSB tehnike se mogu klasifikovati i po veličini. One fiksnih veličina ugrađuju sličan broj bitova poruke u svaki piksel slike nosioca, a one promenljivih veličina za svaki piksel ugrađuju nasumičan broj bitova. Više varijacija je danas dostupno, s različitim nivoima podložnosti detekciji. Jedna od prednosti je direktno ugrađivanje tajne poruke. Međutim, osetljivost na filtere ili promene je velika mana. Procesi poput skaliranja, rotacije, dodavanja šuma ili lossy kompresije stego slike uništiće tajne informacije.

Bez obzira na prednosti metoda, postoje problemi kao što su:

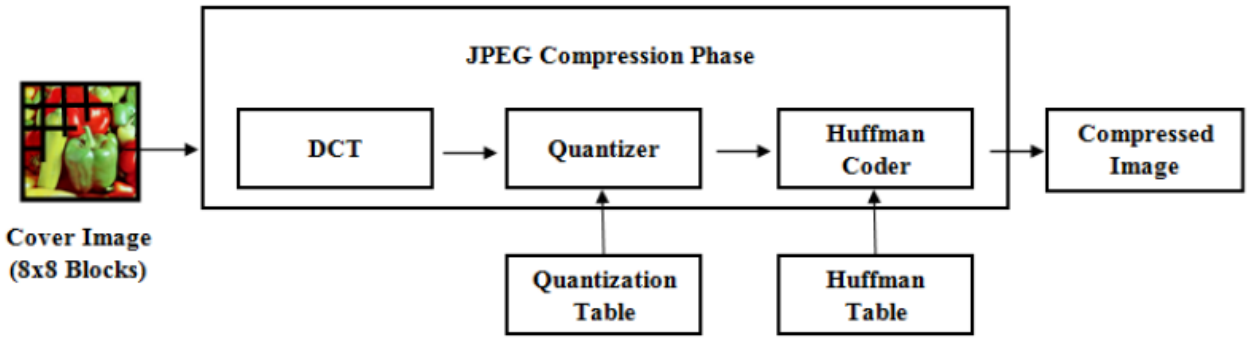
- Direktno korišćenje slike nosioca za ugradnju osetljivih informacija bez enkripcije, što dovodi do lakšeg izvlačenja tajnih poruka ukoliko napadač dođe do algoritma ugradnje
- Upotreba neefikasnih algoritama za ugradnju može proizvesti vizualnu distorziju kod stegoslike, što povećava verovatnoću detekcije ljudskim okom
- Disbalans između kvaliteta slike, kompleksnosti računanja, sadržaja i bezbednosti loš je za upotrebu u realnom vremenu i bezbednosnim primenama.

Otkriće mehanizma ugradnje u najmanje značajne bitove je veliko dostignuće. Iako nije savršeno u prevari ljudskog vizelnog sistema, slaba otpornost na napade dovela je do primena unutar frekventnih domena.

3.1.2. DCT

DCT (*eng. Discrete Cosine Transformation*) je opšta ortogonalna transformacija za digitalnu obradu slika i signala, s prednostima poput visokog nivoa kompresije, malog procenta greške i sposobnosti dobre integracije informacija. DCT dozvoljava da se slika podeli na različite frekventne opsege – više, srednje i niske – tako omogućavajući odabir opsega u koji će se informacije uneti. Uglavnom se bira srednji opseg jer takva ugradnja ne širi informacije na vizuelno najbitnije delove slika, niti ih prekomerno izlaže otklanjanju kroz kompresiju i napade, gde se targetiraju visoki opsezi.

Ovo je najrasprostranjeniji sistem za kompresiju slika s gubicima trenutno i u nastavku je ilustrovan dvodimenzionalnim DCT-om u formi JPEG sistema [1]. Inicijalno, sistem transformiše sliku u YCbCr ravan boja i deli svaku ravan u 8*8 blokove piksela. Celi blokovi se zatim transformišu u DCT koristeći formulu, koja će biti prikazana u poglavlju 3.3. Sledeći korak uključuje kvantizaciju u kojoj se svi DCT koeficijenti dele specifičnim vrednostima, definisanim u kvantizacionoj tabeli, i zaokružuju na najbliži ceo broj. Rezultat su kvantizovani DCT koeficijenti koji se zatim kompresuju kroz entropijski koder, poput Hafmanovog ili aritmetičkog kodera. U fazi dekodiranja JPEG-a, svi DCT koeficijenti se dekvantifikuju. Nakon toga sledi inverzni DCT za rekonstrukciju podataka. Rezultujuća slika je skoro identična originalnoj. Na slici 4 dat je postupak kompresije JPEG-a.



Slika 4. JPEG faza kompresije. [1]

Ugradnja tajnih informacija u digitalne slike upotrebom kompresije s gubicima nije moguća jer ta vrsta kompresije uništava deo informacija u toku procesa. Zato je bitno naglasiti da se algoritam JPEG kompresije deli na fazu s gubicima, koju prati faza bez gubitaka. Faza s gubicima se sastoji od DCT-a i faze kvantizacije, a faza bez gubitaka od Hafmanovog kodiranja za dalju kompresiju podataka. Steganografija se odvija između ove dve faze i moguće ju je odraditi različitim tehnikama, što će biti pokazano u poglavlju 3.4. Informacije se ugrađuju na ovaj način, nakon kvantizacije a pre Hafmanovog kodiranja, jer je izazovno otkriti informacije jer su u frekventnom, a ne spacijalnom domenu. Pored toga, ugrađene informacije se mogu povratiti uz neke gubitke ili modifikacije podataka slike.

3.1.3. DFT

DFT (*eng. Discrete Fourier Transformation*) je slična DCT-u, s tim što koristi Furijeovu transformaciju umesto kosinusne, pri čemu se povećava kompleksnost izračunavanja.

Neka se koriste $f(x1, x2)$ za predstavljanje slike dimenzija $N1 * N2$, gde su $x1$ i $x2$ integeri i $0 \leq x1 \leq N1, 0 \leq x2 \leq N2$. U nastavku su date formule za dvodimenzionalni DFT i inverzni DFT, respektivno:

$$F(k_1,k_2)=\sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1}f(x_1,x_2)e^{-\frac{2\pi x_1 k_1}{N_1}-\frac{2\pi x_2 k_2}{N_2}}$$

$$f(x_1,x_2)=\frac{1}{N_1N_2}\sum_{k_1=0}^{N_1-1}\sum_{k_2=0}^{N_2-1}F(k_1,k_2)e^{\frac{2\pi x_1 k_1}{N_1}+\frac{2\pi x_2 k_2}{N_2}}$$

Ovde $f(x1,x2)$ i $F(k1,k2)$ formiraju DFT par. U suštini, teret računanja DFT-a na prirodnoj slici je ogroman. Uopšteno, slike se dopunjavaju nulama pre računanja DFT-a kako bi se izbegao aliasing[1]. Najmanja veličina prozora zavisi od veličine slike i filtera, a kada se slika sempluje frekvencijom od $2fmax$, tada sliku veličine $A*B$ treba dopuniti nulama dok veličina ne bude $2*A*2*B$.

Osnovna svojstva DFT-a su:

1. Translacija – ako se slika translira u prostornom domenu, njena faza će takođe biti pomerena u frekventnom domenu pa imamo:

$$F(k_1,k_2)=\exp[-j(ak_1+bk_2)]\leftrightarrow f(x_1+a,x_2+b)$$

Upotrebom ovog svojstva, energija slike se fokusira na niskofrekventni region, posle translacije, pošto su f i F periodične funkcije, energija slike će se fokusirati na centru slike, što daje prostor za različite operacije na slici u frekventnom domenu, a u isto vreme se slaže s našom vizuelnom percepcijom.

2. Rotacija – ako se slika rotira u prostornom domenu za ugao q, onda će se odgovarajuća slika u frekventnom domenu rotirati takođe za ugao q, što se opisuje sledećim:

$$F(k_1cos\theta-\ k_2sin\theta,k_1sin\theta+\ k_2cos\theta)\leftrightarrow f(x_1cos\theta-\ x_2sin\theta,x_1sin\theta+x_2cos\theta)$$

3.1.4. DWT, IWT

Treba pomenuti kratko i dve vrste wavelet transformacija, DWT (*eng. Discrete Wavelet Transformation*) i IWT (*eng. Integer Wavelet Transformation*). Wavelet je mali talas koji osciluje u vremenskom domenu. DWT je relativno nova i efikasna tehnika, koja se širi zbog upotrebe u JPEG2000 standardu za kompresiju. DFT i DCT se smatraju za *full frame* transformacije, i kao takve, kod njih će bilo koja modifikacija transformacionih koeficijenata uticati na celu sliku, osim ako se DCT implementirao upotrebom block-based metoda. S druge strane, DWT ima lokalnost prostornih frekvencija – ako je signal ugrađen, on će lokalno uticati na sliku. Zato ova

[1] Aliasing podrazumeva pogrešnu identifikaciju frekvencije signala, što dovodi do greške ili distorzije.

transformacija nudi frekventni i prostorni opis slici. Ova transformacija više doprinosi sakrivanju informacija jer odvaja informacije visokih i niskih frekvencija na bazi piksela. Nakon primene DWT-a na sliku, tajne informacije se kriju u odabranim DWT koeficijentima. Waveleti se tek od skoro koriste u obradi slika, i smatra se da im treba manje resursa i da dovode do manje distorzije slike u poređenju sa DFT i DCT transformacijama. Primarno se koriste u obradi slika za redukciju šuma, kompresiju i detekciju ivica. Wavelet toolbox u MATLAB-u, na primer, nudi funkcije za obradu waveleta.

Kako DWT dozvoljava nezavisnu obradu rezultujućih komponenti bez značajne primetne interakcije između njih, očekuje se efikasniji proces neprimetne ugradnje. Međutim, korišćeni filteri (ali i ostali kao DCT, FFT) imaju realne koeficijente. Kada se ulazni podaci sastoje od nizova integera, kao što je reč kod slika, rezultujući filtrirani izlaz se neće sastojati od integera, što ne omogućava savršenu rekonstrukciju originalne slike. Međutim, uvođenjem metode koja mapira integere u integere, izlaz se kompletno može njima i izraziti. Zato je kreiran IWT, koji dovodi do precizne dekompresije originalnih podataka.

3.2. Postojeći softver

Steganografski softver se koristi za izvršenje različitih funkcija kako bi se sakrili podaci – što uključuje kodiranje podataka kako bi se oni pripremili za sakrivanje unutar nekog fajla, pamćenje koji bitovi fajla nosioca sadrže skrivene podatke, enkriptovanje podataka i ekstraktovanje istih od strane primaoca kome je fajl namenjen. Postoje vlasnički ali i programi otvorenog koda za steganografiju, kao npr. OpenStego, Xiao Steganography, Crypture, itd. Primeri alata koji koriste LSB steganografiju su S-Tools i Hide and Seek. Nosioci alata uključuju BMP, GIF i PNG slike u lossless formatu, a LSB steganografija se može koristiti nad grayscale ili 24-bitnim slikama. Jsteg, OutGuess i F5 koriste DCT metodu.

S-Tools – Posebna metoda koja uključuje modifikaciju LSB-a svake od tri boje piksela na 24-bitnoj slici. S-Tools se prvenstveno koristi za sakrivanje tajnih podataka u BMP i GIF datoteke i upotrebljava različite algoritme šifrovanja kako bi se tajne informacije šifrirale pre ugradnje. Koristi pseudoslučajni broj *m* za zamenu LSB-a, što otežava izdvajanje tajnih podataka. Nakon ubacivanja slike nosioca, sistem će reći korisniku koja je to količina tajnih informacija u bajtovima koje se mogu sakriti na slici.

Hide and Seek -Tehnika koja pokušava da promeni paletu kako bi se ugradile tajne informacije. Ovaj alat pretvara tabelu boja iz 256 boja u 128, a zatim sprovodi umnožavanje svake od boja. U tabeli boja susedni unosi su međusobni duplikati. Verzija 4.1 traži da slika bude 320*480 piksela i da sadrži 256 boja. U slučaju neusklađene veličine, slika će biti okružena prazninama ili isečena.

Slabost ove tehnike leži u prepoznatljivosti poruke svakog ko odradi skeniranje stegoslike radi nalaženja skrivenih informacija.

OutGuess – Ovde je implementiran proces koji obuvhata dva prolaza kroz podatke, gde se prvi sastoji od ugradnje bitova tajne poruke sa pseudo nasumičnim prolaskom po LSB-ovima DCT koeficijenata, preskačući one koeficijente koji su jednaki nuli. Drugi prolaz uključuje korekcije za vrednosti koeficijenata kako bi se garantovalo da histogram DCT-a stegoslike se podudara s njegovim pandanom kod slike nosioca. OutGuess je prilično logičan, kad god se bit modifikuje za skrivanje informacija, traži se ekvivalentni bit i menja se zbog održavanja balansa statističkog profila. Bez obzira na to, program rezerviše oko polovine koeficijenata kako bi ispravilo statističke devijacije u histogramu globalnih koeficijenata, kao rezultat promena LSB-ova u drugoj polovini. Proces je efikasan u održavanju globalnog histograma koeficijenata, iako smanjuje kapacitet za oko polovine originalnog kapaciteta.

Jpeg-Jsteg – Nadaleko poznat alat razvijen još 1997.godine koji menja LSB-ove kvantifikovanih DCT koeficijenata tajnim informacijama i izbegava sve koeficijente s vrednostima 0,1 ili -1. Kapacitet ugradnje je jako ograničen – u najboljem slučaju dolazi do 12% veličine slike. Takođe se može lako detektovati, čak i pri prilično niskim stopama ugradnje kroz napad na histogram. Generalno, koeficijenti JPEG kompresije čine zvonastu krivu, koju proces sakrivanja informacija narušava. Štaviše, JSteg-u nije potreban tajni ključ, tako da će svako ko je upoznat sa steganografskim sistemom moći da povrati skrivene informacije.

F5 – Metoda za ugradnju tajnih podataka u JPEG slike. Umesto da okrene LSB-ove za kodiranje bitova poruka, F5 smanjuje apsolutnu vrednost DCT koeficijenta za jedan, kroz postupak koji se naziva kodiranje matrice (ili ugrađivanje matrice), a koji može smanjiti broj steganografskih modifikacija i održati histogram koeficijenata koji se pojavljuju nenarušenim. Ovaj algoritam se ne može otkriti napadom histograma (χ^2 test) jer ugrađivanje ne zavisi od razmene parova vrednosti. Međutim, otkriveno je da ovaj algoritam i dalje modifikuje histograme koeficijenata na način koji može biti detektovan. Proces ugrađivanja povećava broj nula što dovodi do skupljanja histograma koeficijenta do centra. Algoritam F5 inicijalno rekompresuje sliku sa korisničkim faktorom kvaliteta, a zatim se DCT koeficijenti koriste za ugrađivanje poruke koja ne bi trebalo da bude veća od 14% veličine slike nosioca da se ne bi otkrila vizuelnom analizom.

3.3. JPEG

JPEG je često korišćen metod lossy kompresije za digitalne slike, posebno za digitalne fotografije. Stepen kompresije se može podesiti, dozvoljavajući kompromis između veličine i kvaliteta slike. JPEG tipično postiže 10:1 kompresiju bez vidljivog gubitka kvaliteta. Od svog uvođenja 1992. godine, postao je najčešće korišćen standard i format slike u svetu, sa više biliona kreiranih slika svakog dana počev od 2015. godine. 2000. godine uveden je i format koji je trebalo da bude naslednik, JPEG 2000, ali on nije uspeo da zameni JPEG kao dominantni standard.

JPEG kompresioni algoritam najbolje funkcioniše kod fotografija i slika realističnih scena s glatkim prelazima tonova i boje. JPEG nije dobar za upotrebu kod crtanja linija i ostalih tekstualnih objekata, gde postoji oštar kontrast između susednih piksela. Takve slike bolje je pamtiti u lossless formatima: TIFF, GIF, PNG. Budući da, kao lossless metoda, smanjuje vernost slike, JPEG nije prikladan za preciznu reprodukciju podataka sa slike (te se ne koristi u naučne ili medicinske svrhe). JPEG takođe nije pogodan za fajlove nad kojima se vrše višestruka editovanja, jer se kvalitet gubi svaki put kada se slika ponovo kompresuje. Za sprečavanje gubitka informacija prilikom sekvencijalnog ili ponavljanog editovanja, prvi edit može se sačuvati u lossless formatu, a tek na kraju svih izmena vratiti u JPEG za dalju distribuciju.

Iako se JPEG može kreirati (kodirati) na više načina, najčešće je to JFIF kodiranjem, koje se sastoji iz sledećih koraka:

1. Reprezentacija boja slike pretvara se u YCbCr format, koji se sastoji od jedna luma komponente (Y) koja predstavlja svetlinu, i dve hroma komponente (Cb i Cr) koje predstavljaju boju. Ovaj standard se koristi u digitalnoj televiziji i dozvoljava veću kompresiju bez značajnog efekta na perceptivni kvalitet slike. Kompresija je efikasnija jer su informacije o osvetljenosti, koje su bitnije za percpetivni kvalitet slike, ograničene na samo jedan kanal. Konverzija u YCbCr je specificirana JFIF standardom kao u nastavku. Treba napomenuti da se vrednosti iz RGB formata kreću u opsegu [0, 255], a u tom opsegu ostaju i rezultati YCbCr formata.

$$Y' = 0.299 * R + 0.587 * G + 0.114 * B$$
$$Cb = 128 - (0.168736 * R) - (0.331264 * G) + (0.5 * B)$$
$$Cr = 128 + (0.5 * R) - (0.418688 * G) - (0.081312 * B)$$

   Međutim, u nekim JPEG implementacijama ovaj korak se ne primenjuje, već se informacije o boji zadržavaju u RGB modelu, gde se komponente čuvaju u posebnim kanalima. Ovo dovodi do manje efikasne kompresije i ne koristi se kada je veličina fajla bitan faktor.

2. Rezolucija hroma podataka se smanjuje, uglavnom 2 ili 3 puta (tzv. *downsampling*). Zbog gustine receptora osetljivih na boju i osvetljenost u ljudskom oku, ljudi mogu videti znatno

sitnije detalje u osvetljenosti slike (Y' komponenta) nego u nijansi i zasićenosti boje slike (Cb i Cr komponenta). Odnosi koji se obično koriste su 4: 4: 4 (nema smanjenja), 4: 2: 2 (smanjenje za faktor 2 u horizontalnom smeru) ili 4: 2: 0 (smanjenje za faktor 2 u horizontalnom i vertikalnom smeru), što se najčešće i koristi. U nastavku procesa, komponente se obrađuju odvojeno, ali na vrlo sličan način.

3. Nakon downsample-ovanja, svaki kanal mora biti podeljen na blokove veličine 8*8. Ako podaci za kanal ne predstavljaju celobrojni broj blokova, koder mora ispuniti preostalo područje nepotpunih blokova nekim oblikom lažnih podataka, što se vrši na različite načine. Dalje, svaki 8*8 blok svake komponente (Y, Cb, Cr) predstavlja se u frekventnom domenu, koristeći normalizovanu dvodimenzionalnu diskretnu kosinusnu transformaciju tipa II (DCT). Pre izračunavanja DCT bloka, njegove vrednosti se pomeraju iz pozitivnog opsega u opseg centriran na nulu. Za 8-bitnu sliku, svaki unos u originalnom bloku spada u opseg [0..255]. Srednja tačka opsega (u ovom slučaju vrednost 128) oduzima se od svakog unosa da bi se dobio opseg podataka koji je centriran na nulu, tako da je izmenjeni opseg [-128, 127]. Ovaj korak smanjuje zahteve za dinamičkim opsegom u fazi obrade DCT koja sledi. Dvodimenzionalna transformacija data je sledećom formulom:

$$G_{u,v} = \frac{1}{4}\alpha(u)\alpha(v)\sum_{x=0}^{7}\sum_{y=0}^{7} g_{x,y}\cos\left[\frac{(2x+1)u\pi}{16}\right]\cos\left[\frac{(2y+1)v\pi}{16}\right]$$

Ovde u i v predstavljaju horizontalnu i vertikalnu prostornu frekvencu (koja za blok uzima vrednosti od 0 do 7), $g_{x,y}$ je vrednost piksela u koordinatama (x, y), a α_u je faktor normalizacije, koji može uzeti vrednost $1/\sqrt{2}$ ako je u jednako nuli, a u suprotnom uzima vrednost 1. $G(u, v)$ je rezultujući DCT koeficijent za piksel na koordinati (u, v). Na slici 5 je dat primer jednog bloka 8*8 DCT koeficijenata.

$$G = \begin{bmatrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \downarrow v. \end{matrix}$$
$$\overset{u}{\longrightarrow}$$

*Slika 5. DCT koeficijenti jednog 8*8 bloka. [11]*

Može se primetiti velika vrednost koeficijenta u gornjem levom uglu. Ovo je DC koeficijent (takođe se naziva i konstantna komponenta), koji definiše osnovnu nijansu za ceo blok. Preostalih 63 koeficijenta su AC koeficijenti (*eng. Alternating Components*). Prednost DCT-a je njegova tendencija da sakuplja veći deo signala u jednom uglu rezultata. Korak

kvantizacije koji sledi naglašava ovaj efekat, istovremeno smanjujući ukupnu veličinu DCT koeficijenata, što rezultira signalom koji se lako efikasno kompresuje u fazi entropije.
DCT privremeno povećava dubinu bita podataka, jer DCT koeficijenti 8-bitne komponente slike zauzimaju do 11 ili više bitova (u zavisnosti od vernosti DCT proračuna) za skladištenje. Ove vrednosti se u koraku kvantizacije obično smanjuju na 8-bitne vrednosti.

4. Vrednosti frekventnih komponenti prolaze kvantizaciju. Ljudsko oko je dobro u uočavanju malih razlika u osvetljenosti na relativno velikom području, ali ne tako dobro u razlikovanju tačne jačine varijacije osvetljenosti visoke frekvencije. Zato se jačine visokofrekventnih komponenti skladište s manjom preciznošću od niskofrekventnih komponenti. To se postiže jednostavnim deljenjem svake komponente u frekvencijskom domenu konstantom za tu komponentu, a zatim zaokruživanjem na najbliži celi broj. Ova operacija zaokruživanja jedina je operacija sa gubicima u čitavom procesu (osim downsample-ovanja hroma komponenti) ako se izračunavanje DCT izvodi sa dovoljno visokom preciznošću. Kao rezultat ovoga, tipičan je slučaj da su mnoge komponente više frekvencije zaokružene na nulu, a mnoge od ostalih postaju mali pozitivni ili negativni brojevi, kojima treba mnogo manje bitova. Elementi u matrici kvantizacije kontrolišu odnos kompresije, a veće vrednosti proizvode veću kompresiju. Tipična matrica kvantizacije (za kvalitet od 50% kako je navedeno u originalnom JPEG standardu) data je na slici 6.

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

Slika 6. Tipična matrica kvantizacije. [11]

Koristeći ovu matricu, primenjuje se sledeća formula:

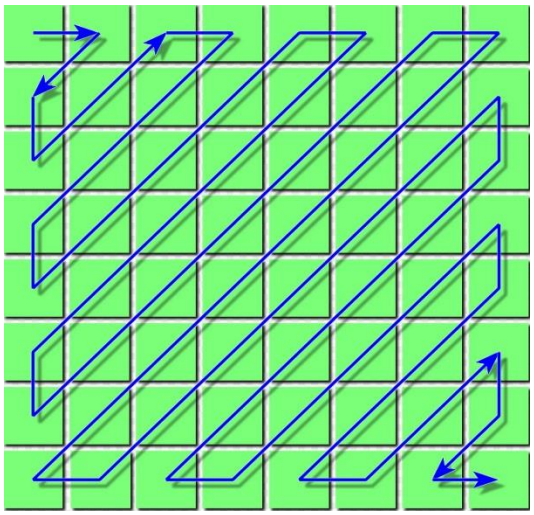
$$B_{j,k} = round\left(\frac{G_{j,k}}{Q_{j,k}}\right), za\ j = 0,1,2,..7\ ; k = 0,1,2,..7$$

G su nekvantizovani DCT koeficijenti, Q je matrica kvantizacije, a B su generisani kvantizovani koeficijenti. Primenom matrice sa slike 6 na blok sa slike 5 dobijamo kvantizovane koeficijente, predstavljene na slici 7. Može se uočiti da je većina višefrekventnih elemenata bloka (onih sa prostornom frekvencom većom od 4) kvantizovana u nule.

$$B = \begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Slika 7. Kvantizovani DCT koeficijenti. [11]

5. Rezultujući podaci 8*8 blokova dalje prolaze kroz kodiranje entropije. To je poseban oblik kompresije podataka bez gubitaka. Uključuje raspoređivanje komponenata slike u „cik-cak“ redosledu koristeći RLE algoritam (*eng. Run Length Encoding*) koji grupiše slične frekvencije, ubacujući nule za kodiranje dužine, a zatim koristi Hafmanovo kodiranje na onom što je ostalo. JPEG standard takođe omogućava, ali ne zahteva, podršku za korišćenje aritmetičkog kodiranja, koje je matematički superiornije od Hafmanovog kodiranja. Prethodni kvantizovani DC koeficijent koristi se za predviđanje trenutnog kvantizovanog DC koeficijenta. Razlika između njih je kodirana, a ne stvarna vrednost. Kodiranje 63 kvantizovanih AC koeficijenta ne koristi takvo razlikovanje predviđanja. Cik-cak sekvenca na 8*8 bloku prikazana je na slici 8. Ovo kodiranje može biti sekvencijalno i progresivno. Kod sekvencijalnog, kodiraju se koeficijenti jednog po jednog bloka, dok progresivno kodiranje radi sa slično pozicioniranim skupom koeficijenata svih blokova odjednom (skeniranje), sve dok ne obradi sve koeficijente svih blokova.



*Slika 8. Cik-cak sekvenca za RLE kodiranje nad blokom dimenzija 8*8.*

6. Kako bi se kodirao generisani obrazac koeficijenata, JPEG koristi Hafmanovo kodiranje. JPEG standard ima Hafmanove tabele za opštu namenu, ali se mogu i generisati tabele optimizovane za određene frekvence slika koje će se kodirati. Proces kodiranja počinje RLE-om, gde je x kvantizovani AC koeficijent čija vrednost nije nula, *RUNLENGTH* je broj nula pre x, *SIZE* je broj bitova neophodan za predstavljanje x, a *AMPLITUDE* je reprezentacija x-a u bitovima. Uz ispitivanje svakog AC koeficijenta koji nije nula kreiraju se dva simbola, prikazana u tabeli.

Symbol 1	**Symbol 2**
(RUNLENGTH, SIZE)	(AMPLITUDE)

RUNLENGTH i *SIZE* čine jedan bajt, što znači da svaki sadrži 4 bita informacija. Viši bitovi predstavljaju broj nula, a niži broj bitova neophodni za kodiranje vrednosti x. Ovo znači da će simbol1 moći da pamti informacije o prvih 15 nula koje prethode nenultom AC koeficijentu. Međutim, JPEG definiše dve specijalne kodne reči. Kodna reč (15,0)(0) se koristi za niz nula koji je duži od 15 pre nego što dođe do nenultog AC-a, a kodna reč (0,0) predstavlja preuranjeni kraj sekvence kada su preostali koeficijenti nula (End of Block, EOB). Ceo proces se izvršava dok se ne dostigne EOB.
Kvantizovani koeficijenti sa slike 8 će nakon ovog procesa dati sledeću sekvencu:

(0, 2)(-3); (1, 2)(-3); (0, 1)(-2); (0, 2)(-6); (0, 1)(2); (0, 1)(-4); (0, 1)(1); (0, 2)(-3); (0, 1)(1); (0, 1)(1); (0, 2)(5); (0, 1)(1); (0, 1)(2); (0, 1)(-1); (0, 1)(1); (0, 1)(-1); (0, 1)(2); (5, 1)(-1); (0, 1)(-1); (0, 0).

Može se primetiti da se prvi koeficijent (-26) kao DC koeficijent ne koristi u ovom kodiranju. Odavde se računanje bazira na broju pojavljivanja koeficijenata. U primeru, većina kvantizovanih koeficijenata su mali brojevi kojima ne prethode nulti koeficijenti. Hafmanov algoritam će generisati najefikasnije binarno kodno stablo za datu distribuciju pojavljivanja. Najčešći slučajevi biće predstavljeni kraćim kodnim rečima. Generisani kodovi su prefiksni kodovi – nijedan dodeljen kod nije prefiks nekom drugom dodeljenom kodu. Generalno, postoje već definisane Hafmanove tabele za JPEG format, ali nije ih nužno koristiti – neće svaka biti podjednako delotvorna za svakog nosioca. One se mogu preračunavati i dinamički.

Proces dekodiranja poništava sve korake – nakon dekompresije i dekvantizacije istom matricom kvantizacije, koristi se inverzna DCT (IDCT, tip III po standardu), čija je formula data u nastavku.

$$f_{x,y} = \frac{1}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} \alpha(u)\alpha(v) F_{u,v} \cos\left[\frac{(2x+1)u\pi}{16}\right] \cos\left[\frac{(2y+1)v\pi}{16}\right]$$

Nakon vraćanja vrednosti u opseg [0-255] dodavanjem vrednosti 128 svakom elementu matrice, konačno se vrši transformacija iz YCbCr u RGB model boja.

3.4. Predložena rešenja

Kada se JPEG kodiranje koristi u steganografiji, kao što je već napomenuto u poglavlju 3.1.2, skrivanje podataka se uglavnom odvija nakon DCT-a, a pre kodiranja – dakle, u fazi kvantizacije. Koristi se taj momenat jer se podaci tada nalaze u frekventnom, a ne u spacijalnom domennu. Postoje različiti načini za ugrađivanje podataka i dosta naučnih radova se bavi upotrebom DCT steganografije i različitih metoda skrivanja podataka u JPEG formatu. U nastavku je dato samo par takvih metoda.

Sachnev i Kim [12] su 2008. godine predložili metodu koja uklanja i ubacuje AC koeficijente čije su vrednosti 1 ili -1. Kandidati se traže među svim AC koeficijentima nakon kvantizacije. Predložena tehnika označava najpogodnije koeficijente, koji će pružiti manju degradaciju nosioca nakon modifikacije. Poređen je sa F5 i OutGuess-om nad kojima je izvršena stegoanaliza i dao je bolje rezultate – manje se mogao detektovati. Sekvenca DCT koeficijenata nudi određene informacije o slici. Predložena šema i dodaje i uklanja koeficijente – suštinski, ona samo menja raspored u nizu DCT koeficijenata, ne menja značajno koeficijente.

Iste godine je predložena i metoda koja se sprovodi na kvantizovanim DCT koeficijentima iz YUV prostora boja [13]. Tri koeficijenta iste frekvencije čine triplet kao nosilac poruke, pa se na osnovu svojstva tripleta određuje njegov kapacitet ugradnje. Za ugradnju se koriste mod2 i mod4 aritmetičke operacije, a kada je neophodna modifikacija, generiše se slučajan broj za smer obrtanja i poboljšana SRM (*eng. Shortest Route Modification*) šema za smanjenje distorzije. Metoda ima visok kapacitet ugradnje, dok približno čuva histogram kvantizovanih DCT koeficijenata.

Leng, Labadin i Juan su implementirali tehniku koja skriva informacije u frekventnom domenu uz pridodatu tehniku reparacije [14]. Na ovaj način se ispravljaju bilo koje statičke devijacije na nosiocu nakon ugradnje skrivene poruke. Tehnika treba da izdrži vizualne i statističke napade pa se može koristiti da pojača bezbednost steganografskog sistema. Tehnika je prvenstveno namenjena JPEG fajlovima ali bi se mogla primenjivati i u ostalim domenima. Primenjuje se kada se poruka ugrađuje LSB tehnikom u frekventnom domenu.

Berres i Soria-Lorente su kreirali steganografsku tehniku zasnovanu na kompresionom standardu JPEG formata i tehnici entropijskog praga (*eng. Entropy Thresholding*) [15]. Algoritam koristi jedan javni i jedan privatni ključ za generisanje binarne sekvence nasumičnih brojeva koji pokazuju gde se elementi binarne sekvence tajne poruke smeštaju. Zatim se oni unose u prvih sedam AC koeficijenata u transformisanom DCT domenu. Naravno, pre unošenja poruke, slika je podvrgnuta različitim transformacijama. Do unošenja podataka dolazi samo ukoliko je entropijski prag određenog bloka zadovoljen i ukoliko slučajno odabran broj to pokazuje. Autori su proveravali rad algoritma računanjem PSNR-a, metrika distorzije i analizama histograma.

Kod [9] je primenjena kombinacija DCT-a uz upotrebu SSB-4 tehnike spacijalnog domena. Glavna ideja je koristiti četvrti bit DCT koeficijenta nosioca za sakrivanje bitova poruke, uz zadržavanje minimalne razlike, eventualnim menjanjem prvog, drugog trećeg i/ili petog bita koeficijenta. Ako je vrednost četvrtog bita i bita poruke jednaka, ne radi se ništa, u suprotnom se bit menja bitom poruke i modifikuju se ostali bitovi po definisanom algoritmu kako bi se smanjila razlika. PSNR pokazuje da je klasičan DCT LSB bolji ali i ima više šanse da izgubi podatke prilikom kompresije i da ga otkriju stegoanalizom, posebno jer nije osetljiv na različite vrste slika.

2017. godine predložena je reverzibilna šema skrivanja podataka u JPEG slike, konkretno, u kvantizovane koeficijente jednake nuli [16]. Par susednih takvih nula u skupu DCT blokova koristi se za ugradnju podataka. Reverzibilna šema podrazumeva da se u procesu ekstrakcije slika nosilac takođe rekonstruiše potpuno, a ne samo poruka. Prvo algoritam bira par pozicija iz skupa u zavisnosti od prvog dela tajne poruke, i menja koeficijente malom razlikom u zavisnosti od drugog dela poruke. Rezultati pokazuju da metoda ima visok kapacitet ugradnje i nizak odnos između povećane veličine fajla i payloada.

U radu koji su predložili Attaby i Ahmed, fokus je na načinu samog skrivanja poruke [17]. Skrivena poruka prlazi kroz dve faze kompresije pre ugradnje: u prvoj fazi, poruka se kompresuje uklanjanjem „slabih“ reči i zamenom izraza čestim skraćenicama. U drugoj fazi, rezultat se dalje kompresije Hafmanovim algoritmom. Nakon toga, poruka se ugrađuje u nosioca na osnovu modula 3 razlike između DCT koeficijenata nosioca tokom JPEG faze kompresije. Tehnika je nazvana DCT-M3. Najviše steganografskih tehnika se fokusira na delove slike gde će se poruka kriti a zanemaruje se način skrivanja. Kao rezultat, uglavnom se koristi LSB, što povećava procenat manipulacije na nosiocu. Ova tehnika je efikasnija i manje manipuliše nosiocem od standardnog LSB-a.

Većina radova krije jedan bit u svaki odabrani koeficijent, gde se oni kriju direktnom modifikacijom – primenom LSB metode ili indirektnim menjanjem vrednosti koeficijenata, npr. menjanjem znaka bita koeficijenta. U [18] se umesto direktne ugradnje primenjuje indirektni pristup za sakrivanje po dva bita tajne poruke u odabrane DCT koeficijente. Kao i inače, modifikovani koeficijenti se dalje kompresuju entorpijskim kodiranjem. Kako bi se povećao kapacitet ugradnje, povećan je broj nenultih kvantizovanih DCT koeficijenata korišćenjem modifikovane kvantizovane tabele, ali i unošenjem po dva bita tajne poruke u svaki dozvoljeni

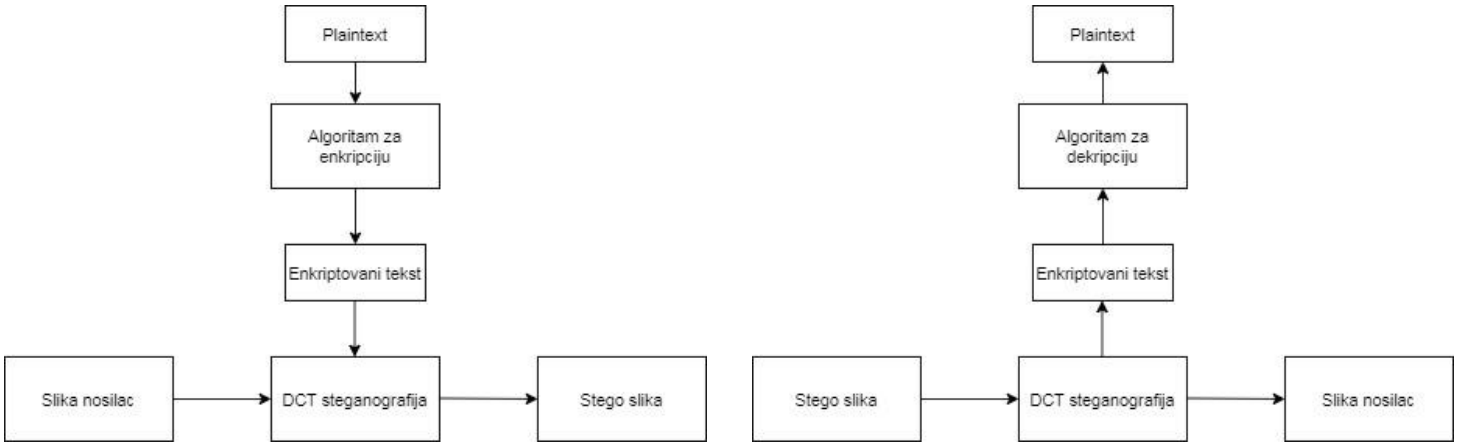
kvantizovani DCT koeficijent slike nosioca. Autori modifikuju predefinisanu kvantizovanu tabelu upotrebom tzv. faktora skaliranja.

Postojeći radovi o steganografiji uglavnom pretpostavljaju da slika ostaje nepromenjena prilikom transmisije od pošiljaoca do primaoca. Ova pretpostavka možda nije toliko logična u eri interneta, zbog nepoznatih kompresija servis provajdera. Zbog toga se skrivene informacije možda ne mogu korektno povratiti na strani primaoca. Jedan od najnovijih radova iz 2020.godine opisuje JPEG steganografski metod koji se može odupreti kompresiji tokom mrežne transmisije, i bez znanja o kom je kompresionom procesu reč. Metoda koristi simuliranu repetitivnu kompresionu mrežu, i u zavisnosti od rezultata izvodi modifikacije nad DCT koeficijentima koje je proces kompresije izmenio. Iz stegoslika generisanih ovom metodom uspešno se mogu izvući originalne tajne poruke.

3.5. Enkripcija

U prethodno prikazanim radovima koji su obrađivali određene steganografske metode, fokus je uglavnom bio na nalaženju pravih podataka slike gde bi se tajna poruka mogla ugraditi. Međutim, neophodno je obratiti pažnju i na sigurnost poruke koja se ugrađuje. Čak i ukoliko steganografska metoda deluje „neprobojno“, uvek postoji šansa da će se nekim od novijih tehnika stegoanalize ona moći „razbiti“, pa bi se na taj način došlo i do tajne poruke. Zato je dobra praksa enkriptovati poruku pre ugradnje. Ukoliko se stegoanalizom otkrije postojanje tajne poruke, bez ključa za enkripciju sadržaj poruke neće biti otkriven.

Za šifriranje se može koristiti neki od simetričnih ili asimetričnih šifratora. Jako je bitno da primalac i pošiljalac imaju tajni ključ (idealno bi bilo da je on njima poznat pre početka procesa, kako se ne bi slao preko mreže i tako bio potencijalno ugrožen), kako bi poruka bila korektno šifrovana i dešifrovana. Najčešće se koriste DES i AES standardi za enkripciju. Koji god šifrator da se koristi, njegova upotreba u steganografskom sistemu bi izgledala kao na slici 9. Jedan od radova [20] pominje DCT stegnografiju upotrebom LSB metode uz kriptovanje poruke *blowfish* algoritmom, koji je sličan AES-u.



Slika 9. Upotreba enkripcije u steganografskom sistemu.

3.6. Opis implementacije

U praktičnom delu implementirana je DCT steganografija ugradnjom tajne poruke u frekventnom domenu, u kvantizovane DCT koeficijente. Korisnik bira sliku nosioca, nakon čega se unosi tekst za sakrivanje. Po ugradnji poruke, prikazaće se stegoslika i biće data opcija otkrivanja tajne poruke. Ta opcija je data kako bi se proverilo da se cela poruka može izvući u originalu. U nastavku su pomenute vrednosti koje se vezuju za test sliku Lena, čije su dimenzije 440 x 439 piksela. Nosilac je prikazan na slici 11.

Kako bi se povećala sigurnost i kako bi se povećao kapacitet ugradnje, data je opcija enkriptovanja poruke koja se ugrađuje AES šifratorom, koji je pomenut u poglavlju 3.5, nakon čega će se vršiti i kompresija Hafmanovim kodom. Ovaj korak nije neophodan, ali je poželjan jer omogućava veću sigurnost ukoliko se stegoanalizom otkrije postojanje poruke. Za AES enkripciju je neophodno generisanje inicijalizacionog vektora i ključa, što je olakšano upotrebom postojeće klase *Aes*. Moguće je ugraditi poruku i kao plaintext, pri čemu ona svakako prolazi kroz Hafmanovu kompresiju. Ukoliko se utvrdi da se tekst ne može ugraditi zbog nedovoljnog kapaciteta ugradnje nosioca, program će obavestiti korisnika.

Treba napomenuti da aplikacija nakon sakrivanja prikazuje nosioca i stegosliku jednu pored druge. Već je rečeno da je za steganografiju bitno da se slika nosilac ne otkriva, kako bi se otežala stegoanaliza – primalac treba da vidi samo stegosliku. Ukoliko bude imao i nosioca, može ih upoređivati. Ovde se ove dve slike vide u isto vreme kako bi se moglo pokazati da, pored toga što na stegoslici nema vidljivih naznaka sakrivene poruke, ona se i ne razlikuje previše od nosioca.

Svi koraci koji su opisani u poglavlju 3.3 jesu obavezni kod JPEG kompresije ali ne moraju nužno koristiti iste vrednosti, što će biti navedeno i u nastavku. Ovde je implementiran postupak po uzoru na JPEG kompresiju – neke vrednosti i koraci su promenjeni. Na samom početku se nosilac konvertuje iz RGB u YCbCr format boja. Dobijaju se tri odvojena kanala (3 niza *float* vrednosti), koji će se nadalje odvojeno modifikovati. Nakon ovoga se kanali Cb i Cr mogu, ali i ne moraju downsample-ovati jer nemaju toliki uticaj na percepciju ljudskog oka. Sledeći korak je deljenje kanala na blokove veličine 8*8, kao pripremu za diskretnu kosinusnu transformaciju. Budući da se unapred ne znaju dimenzije nosioca, a pošto on može biti različitih dimenzija, prvo se dimenzije unetog nosioca proveravaju. Ukoliko neka od dimenzija nije odgovarajuća (nije deljiva sa 8), slika se dopunjuje određenim brojem piksela, ponavljajući najbliže piksele toj ivici. Nakon toga, svaki kanal se deli na blokove veličine 8*8.

U postupku diskrente kosinusne transformacije neophodno je prvo vrednosti u blokovima dovesti u opseg [-128, 127], nakon čega se za svaki blok kreće s računicom. Iako nije naglašeno kao obavezno, i ulaz i izlaz DCT-a su realne vrednosti, kako bi do ovog momenta kompresija bila bez gubitaka. Ova funkcija je svakako ona koja uzima najviše resursa i vremena za računanje –na nivou bloka, za svaki koeficijent treba izvršiti 65 različitih računanja, a ukupan broj koeficijenata jednak je dimenzijama slike u tri kanala. Za testiranu sliku, Lena, čije su dimenzije 440*439 piksela, bilo je neophodno između 15 i 17 sekundi za izvršavanje DCT transformacije i svega što njoj prethodi.

Kvantizacija se svodi na deljenje dobijenih DCT koeficijenata definisanim matricama kvantizacije. Postoje dve matrice koje se koriste kao standard za kvantizaciju kod JPEG-a – jedna je za Y kanal, a druga za Cb i Cr kanale. One su korišćene i u implementaciji. Ukoliko je neophodno imati veći kvalitet slike, moguće je koristiti matrice po svom nahođenju ili matrice prilagođene nosiocima u koje se ugrađuje. Na slici 10 date su kvantizacione matrice koje se koriste za kanal Y i kanale Cb i Cr.

$$Luminance = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix} \quad Chrominance = \begin{bmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{bmatrix}$$

Slika 10. Korišćene kvantizacione tabele.

Pošto se u JPEG kompresiji kao jedan od koraka predlaže downsample-ovanje Cb i Cr kanala jer je ljudsko oko manje osetljivo na njihove podatke, ovde je iskorišćena ta činjenica da se podaci ugrade upravo u te kanale.

Posmatrajući *chrominance* tabelu, vidi se da će se oni elementi iz kanala podeljeni brojem 99 (koji čini više od polovine tabele) najverovatnije svesti na nulu, što i jeste cilj (svi oni iz opsega -98 do

98 biće nula). U povratku, množenjem matricom *chrominance* tu će ostati nule, međutim to jeste deo gubitaka u ovom procesu i to će dovesti do menjanja piksela u povratnoj slici. Ali, ukoliko je prvobitni koeficijent bio neka mala vrednost, na primer 6, koja je kvantizacijom svedena na nulu, menjanje tog kvantizovanog koeficijenta ugradnjom podataka mogla bi da uvede problem, jer bi npr. menjanjem s nule na jedinicu, u povratnom procesu dobili 1*99 = 99, što proizvodi mnogo veću razliku u odnosu na početni element, 6. Zato se za ugradnju skrivene poruke koristi samo par elemenata ovih blokova – oni koji u povratku neće prikazati preveliku razliku s ugrađenim podacima.

Prvo je neophodno ugraditi dužinu poruke koja se sakriva. Pošto je to 32-bitni podatak, on se deli između prva 4 bloka u kanalima Cb i Cr. Nakon toga, kreće ugradnja same poruke. Podaci se ugrađuju u elemente [0,1], [0,2], [1,0] i [2,0] blokova kanala Cb i Cr, kao i u samo jedan element blokova kanala Y, [0,2]. Unos u kanal Y je odrađen čisto zbog testiranja promene boja i koliko to utiče na našu percepciju slike. Element [0,2] je odabran jer je element u *luminance* matrici na toj poziciji najmanji (10), te će i u povratku pri množenju praviti najmanju grešku.

Već nakon ugrađivanja podataka, moguće je odraditi inverznu kvantizaciju, IDCT i vratiti sliku nazad, kako bi se video rezultat sakrivanja, što je i implementirano. Svakako, da bi primalac mogao da otpakuje sliku i ekstrakuje poruku, neophodno mu je dosta informacija. Zato se dalje nastavlja s kompresijom.

Nakon ugrađivanja poruke u kvantizovane DCT koeficijente, po JPEG standardu, sledi kompresija. To je ovde odrađeno slično kao kompresija skrivene poruke – nije rađena zig-zag transformacija i kreiranje parova na osnovu kojih bi se upotrebila Hafmanova tabela, već su se svi koeficijenti iz sva tri kanala, Y, Cb i Cr, smestila u jedan niz, nad kojim su računate frekvence pojavljivanja određenih vrednosti koeficijenata. Frekvence se čuvaju kao ključ-vrednost parovi. Može se primetiti da 0 kao koeficijent čini više od 50% vrednosti iz kanala, što znači da će joj biti dodeljen najkraći Hafmanov kod. Naravno, kao što je rečeno, DC koeficijenti su izuzeti iz kompresije i kodiranja.

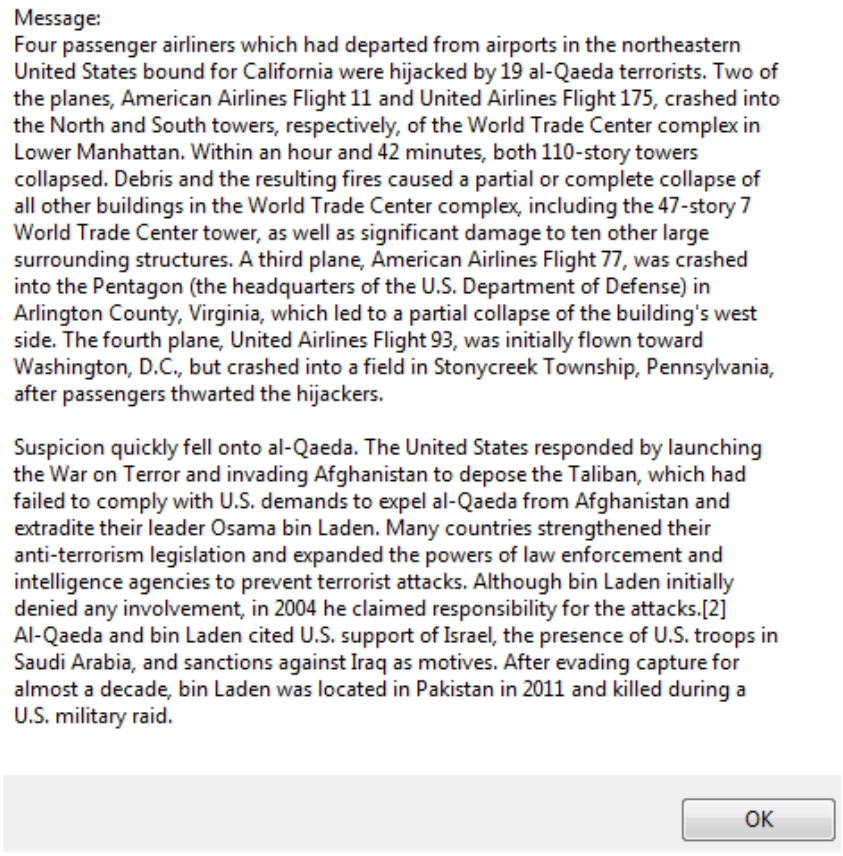
Kada smo dobili kompresovane kanale slike, neophodno je sačuvati sve bitne podatke u novi format. Pri dobijanju fajla, koji se sada ne može otvoriti klasično, primalac treba odraditi inverzne funkcije svega gorepomenutog kako bi mogao da prikaže sliku na ekranu u RGB formatu. Kako bi to odradio, neophodno je da zna originalne dimenzije slike, na osnovu kojih će izračunati i nove dimenzije (proširene da budu deljive sa 8). Uz to, budući da nisu korišćene opšte Hafmanove tabele za kompresiju, treba ugraditi ključ-vrednost parove iz rečnika koji je kreiran prilikom kompresije. Nakon toga, nadovezuju se informacije iz kompresovanih kanala. Oni su konvertovani u niz bitova, kako bi se maksimalno iskoristila kompresija. Naravno, veličina dobijenog formata neće zavisiti samo od jačine kompresije, već i od veličine ugrađenog rečnika, tj. parova ključ-vrednost. Treba u ovaj fajl upisati i padding – kako su kompresovani kanali konvertovani u niz bitova, možda će biti neophodna neka dopuna u poslednjim bitovima podataka kako bi činili ceo bajt. To treba upisati da bi se kasnije ispravno rekonstruisali podaci.

Primalac će ove podatke ekstrakovati iz formata i primeniti sledeće korake, respektivno: dekompresija, dekvantizacija, IDCT, vraćanje na originalne dimenzije, vraćanje na RGB format boja. Ovo je neophodno da bi se slika prikazala na ekranu. Ekstraktovanje poruke vrši se nakon dekompresije a pre dekvantizacije, funkcijom za ekstraktovanje. Neophodno je prvo izvući dužinu poruke, a zatim i samu poruku, pri čemu su oba ova podatka ugrađivana drugačije, te se drugačije i izvlače.

Iako na prvi pogled deluje da kapacitet ugradnje nije veliki, velika količina teksta može stati u sliku, u zavisnosti od njenog formata. U test sliku Lena, čije su dimenzije 440*439 piksela, može stati 27163 bitova tajne poruke (3,32KB). Treba uzeti u obzir da se poruka pre ugradnje kompresuje, tako da stvarna veličina poruke može iznositi i do 5KB, u zavisnosti od toga kolika se ušteda dobija u Hafmanovoj kompresiji.

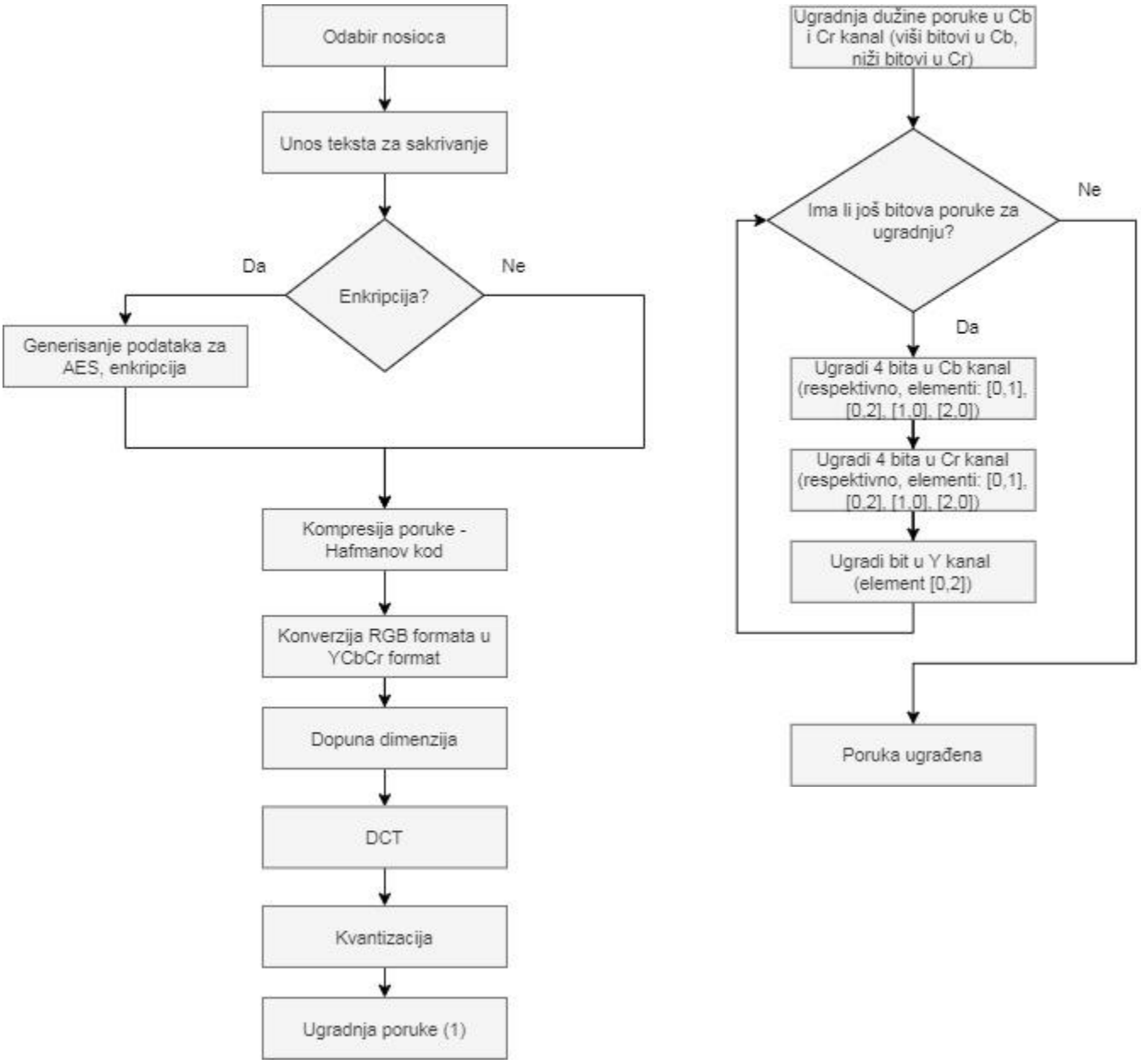
Na slikama 11 i 12 date su test slika nosilac i stego slika u koju je ugrađen tekst od 2494 karaktera, koji je iz nje ekstrakovan i prikazan na slici 13. Stego slika gubi na oštrini u odnosu na nosioca zbog gubitaka koji se i očekuju u procesu kvantizacije tabelama koje garantuju 50% originalnog kvaliteta.

Slike 11 i 12. Nosilac i stego slika, Lena.

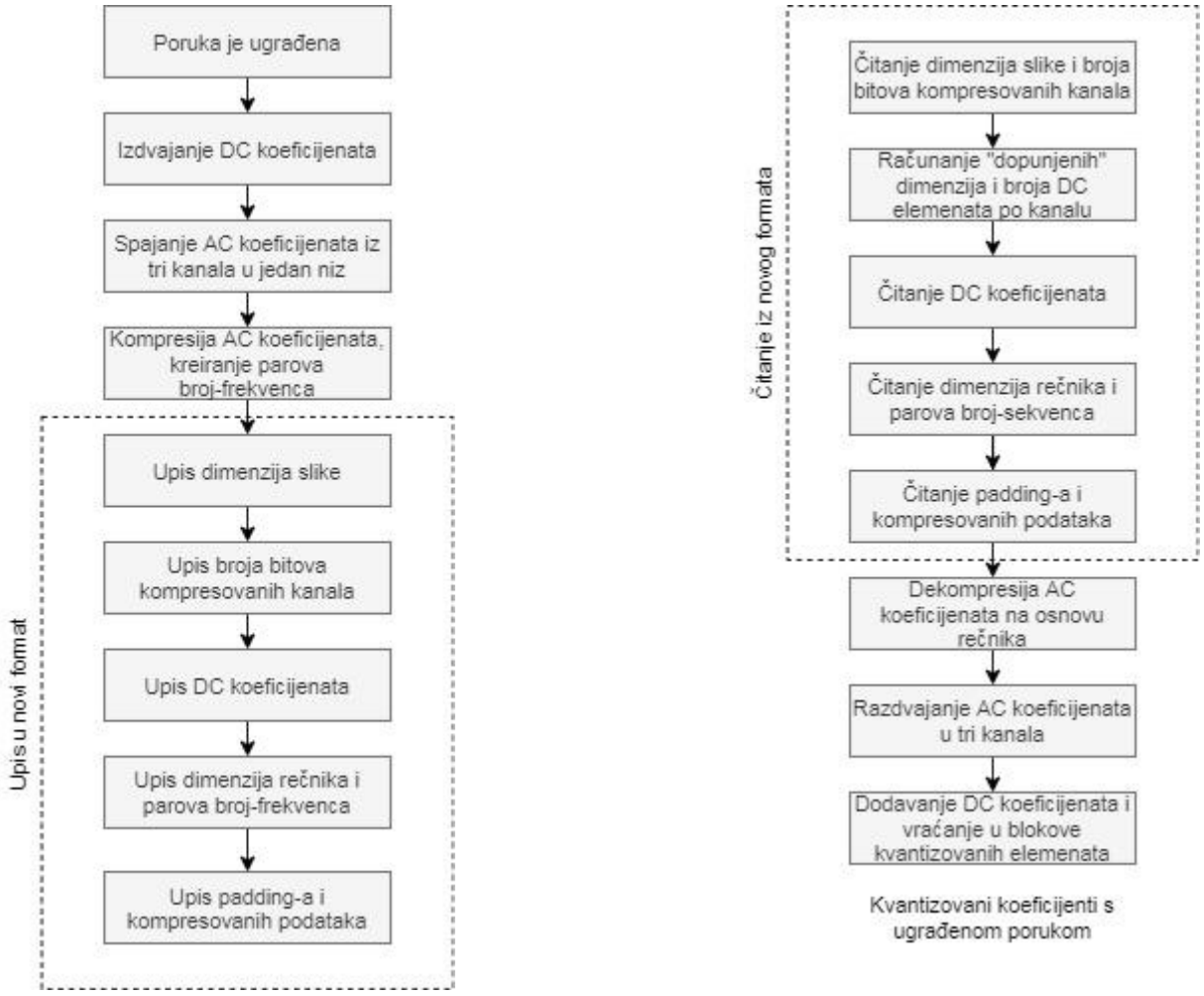


Slika 13. Poruka ekstraktovana iz stegoslike.

Opšti proces unosa nosioca i poruke za skrivanje dat je na slici 14, dok je detaljniji opis same ugradnje dat na slici 15. Ukoliko se odabere čuvanje novog formata, napreduje se s kompresijom podataka i njihovim upisivanjem. Redosled upisivanja podataka je dat na slici 16. Za preuzimanje podataka iz novog formata, prati se dijagram na slici 17.

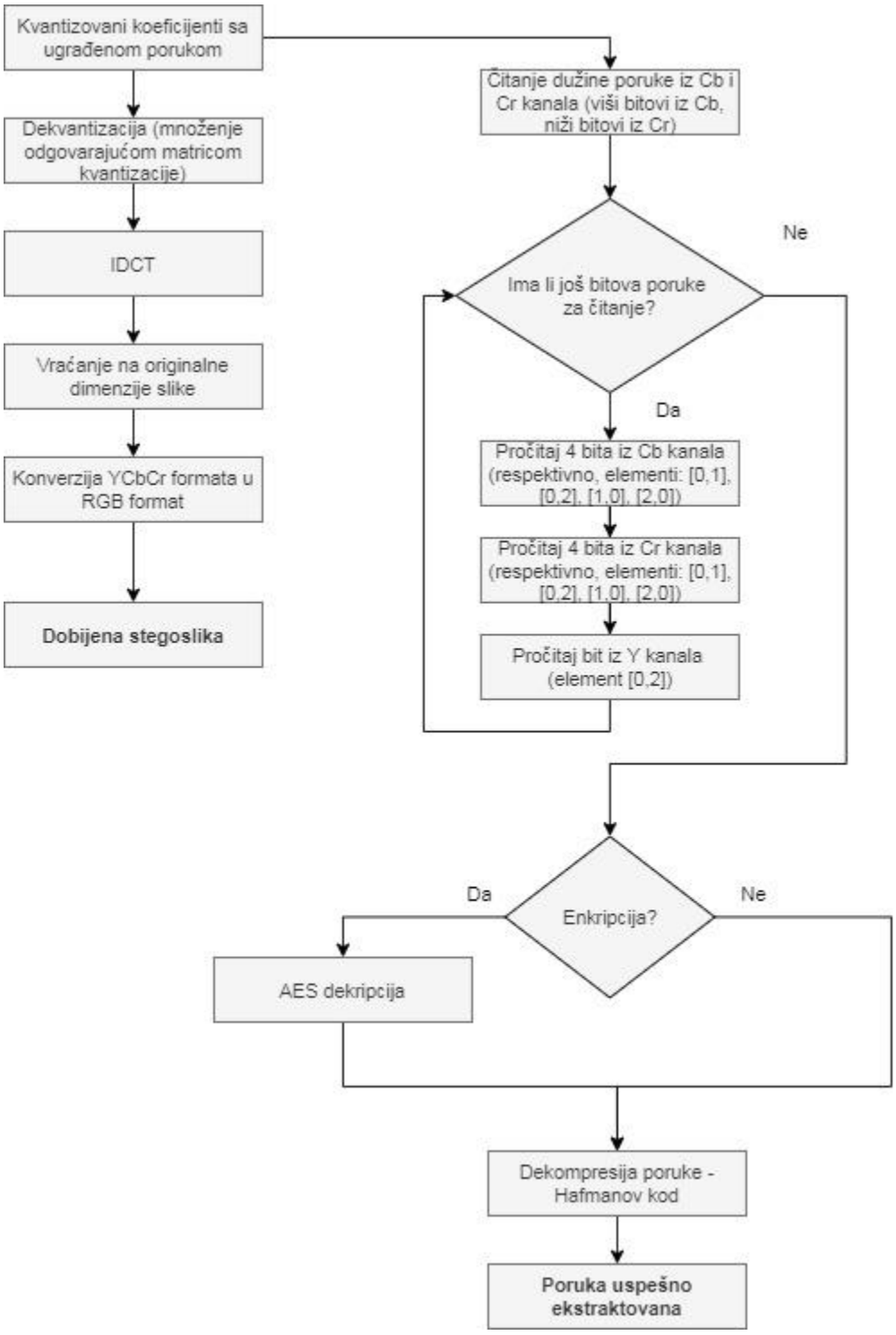


Slike 14 i 15. Unos nosioca, poruke i proces ugradnje poruke.



Slike 16 i 17. Upis i čitanje – novi format. Polazna i krajnja tačka su kvantizovani koeficijenti s ugrađenom porukom.

Kada se dobiju kvantizovani podaci i rekonstruišu kanali, može se prikazati slika s ugrađenim informacijama, a u istom momentu se, ukoliko se zna način ugradnje, informacije mogu ekstrakovati. To je prikazano na slici 18. Nakon ekstrakcije poruke, ona se dekriptuje (ukoliko je korišćen AES) i dekompresuje.



Slika 18. Generisanje stegoslike iz kvantizovanih koeficijenata s tajnom porukom i/ili ekstrakcija poruke.

4. Zaključak

Cilj ovog rada bio je predstavljanje osnovnih koncepata steganografije, uz poseban osvrt na steganografiju slika, s naglaskom na upotrebu DCT metode za sakrivanje podataka. Prvo se govorilo o steganografiji kao pojmu i opštoj podeli, nakon čega je obrađena steganografija slika, kao najrazvijenija grana steganografije. Akcenat je bačen na DCT kao i radove koji koriste neki vid sakrivanja informacija nakon DCT-a. Na kraju je dat opis implementacije koja je uključila i enkripciju podataka, kao dodatni vid zaštite.

Nakon proučavanja glavnih steganografskih algoritama, može se doći do zaključka da ne postoji nijedan univerzalni steganografski metod sa stoprocentnim uspehom. Uvek će se morati kreirati određeni kompromis između robusnosti, kapaciteta i neprimetnosti. Od glavnog cilja primene sakrivanja informacija zavisiće i kojoj će se osobini posvetiti veća pažnja. Povećana upotreba interneta je nesumnjivo dovela do porasta saobraćaja i veliki broj informacija se krije i šalje u različitim formatima. Ovo je oblast koja se konstantno razvija – u poglavlju 3.4. dato je samo par primera radova na ovu temu. Svake godine se implementiraju novi steganografski algoritmi, koje u korak prate novi algoritmi stegoanalize – radi se o trci u kojoj nema pobendika. Svakako, može se zaključiti da u zavisnosti od cilja, neke će karakteristike morati da se na račun drugih žrtvuju.

LITERATURA

[1] S.Atawneh, A. Almomani, P. Sumari, *Steganography in Digital Images: Common Approaches and Tools*, IETE Technical Review, Vol. 30, Iss. 4, 2013.

[2] What is Steganography, SearchSecurity: https://searchsecurity.techtarget.com/

[3] *Hide and Seek: An Introduction to Steganography*, IEEE, 2003.

[4] F.Q. Alyousuf, R. Din, *Review on secure data capabilities of cryptography, steganography and watermark domain*, ResearchGate, 2019.

[5] Steganography, Wikipedia: https://en.wikipedia.org/wiki/Steganography

[6] A. Tiwari, S.R. Yadav, N.K. Mittal, *A Review on Different Image Steganography Techniques*, IJEIT, Vol.3, Iss. 7, January 2014.

[7] J.D.Vico, *Steganography and Steganalysis: Data Hiding in Vorbis Audio Streams*, Master Thesis, Universidad Politecnica di Madrid, 2010.

[8] M.N. Abdulwahed, *An Effective and Secure Digital Image Steganography Scheme Using Two Random Function and Chaotic Map,* ISSN: 1992-8645, 2020.

[9] Nidal M.S. Kafrl, *Bit-4 of frequency domain – DCT steganography technique*, ResearchGate, 2009.

[10] Z. Lu, S. Guo, *Lossless Information Hiding in Images*, Elsevier Inc., 2017.

[11] JPEG, Wikipedia: https://en.wikipedia.org/wiki/JPEG#JPEG_compression

[12] V. Sachnev, H.J. Kim, R. Zhang, Y.S. Choi, *A Novel Approach for JPEG Stegnography,* 7th International Workshop, Korea, 2008.

[13] T. Li, Y. Zhao, R. Ni, L.Yu, *A High Capacity Steganography Algorithm in Color Images* ,7th International Workshop, Korea, 2008.

[14] C.K. Leng, J. Labadin, S. F. Samson Juan, *Steganography: DCT Coefficients Reparation Technique in JPEG Image,* IJDCT vol. 2 no. 2, 2008.

[15] A. Soria-Lorente, S. Berres, *A Secure Steganographic Algorithm Based on Frequency Domain for the Transmission of Hidden Information,* Wiley, 2017.

[16] J Chang, Y. Lee, H. Wu, *Compression-Efficient Reversible Data Hiding in Zero Quantized Coeficients in JPEG Images,* 2017.

[17] A.A. Attaby, M. M. Ahmed, A.K. Alsammak, *Data hiding inside JPEG images with high resistance to steganalysis using a novel technique: DCT-M3*, Ain Shams Engineering Journal, 2017.

[18] A.K. Pal, K. Naik, R. Agarwal, *A Steganography Scheme on JPEG Compressed Cover Image with High Embedding Capacity*, IAJIT, Vol. 16, No. 1, 2019.

[19] F. Li, K. Wu, C. Qin, J. Lei, *Anti-compression JPEG steganography over repetitive compression networks*, Elsevier, 2020.

[20] M. Gunjal, J. Jha, *Image Steganography Using Discrete Cosine Transform and Blowfish Algorithm*, IJCTT, Vol. 11, No. 4, 2014.