

# UTJECAJ VELIČINE BLOKA ZA PROCJENU POKRETA NA KVALITETU INTERPOLIRANE SLIKE U POSTUPKU POVEĆANJA VREMENSKE REZOLUCIJE VIDEO

---

**Marjanović, Mihael**

**Undergraduate thesis / Završni rad**

**2021**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:200:767390>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-02-23**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I  
INFORMACIJSKIH TEHNOLOGIJA**

**Sveučilišni studij**

**UTJECAJ VELIČINE BLOKA ZA PROCJENU  
POKRETA NA KVALITETU INTERPOLIRANE SLIKE U  
POSTUPKU POVEĆANJA VREMENSKE REZOLUCIJE  
VIDEA**

**Završni rad**

**Mihael Marjanović**

**Osijek, 2021.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom sveučilišnom studiju

Osijek, 17.09.2021.

Odboru za završne i diplomske ispite

**Prijedlog ocjene završnog rada na  
preddiplomskom sveučilišnom studiju**

<b>Ime i prezime studenta:</b>	Mihael Marjanović
<b>Studij, smjer:</b>	Preddiplomski sveučilišni studij Računarstvo
<b>Mat. br. studenta, godina upisa:</b>	R4241, 26.07.2018.
<b>OIB studenta:</b>	29935492870
<b>Mentor:</b>	Doc. dr. sc. Denis Vranješ
<b>Sumentor:</b>	
<b>Sumentor iz tvrtke:</b>	
<b>Naslov završnog rada:</b>	Utjecaj veličine bloka za procjenu pokreta na kvalitetu interpolirane slike u postupku povećanja vremenske rezolucije videa
<b>Znanstvena grana rada:</b>	<b>Telekomunikacije i informatika (zn. polje elektrotehnika)</b>
<b>Predložena ocjena završnog rada:</b>	Vrlo dobar (4)
<b>Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:</b>	Primjena znanja stečenih na fakultetu: 2 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 2 bod/boda Razina samostalnosti: 2 razina
<b>Datum prijedloga ocjene mentora:</b>	17.09.2021.
<b>Datum potvrde ocjene Odbora:</b>	22.09.2021.
Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 25.09.2021.

**Ime i prezime studenta:**

Mihael Marjanović

**Studij:**

Preddiplomski sveučilišni studij Računarstvo

**Mat. br. studenta, godina upisa:**

R4241, 26.07.2018.

**Turnitin podudaranje [%]:**

4

Ovom izjavom izjavljujem da je rad pod nazivom: **Utjecaj veličine bloka za procjenu pokreta na kvalitetu interpolirane slike u postupku povećanja vremenske rezolucije videa**

izrađen pod vodstvom mentora Doc. dr. sc. Denis Vranješ

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

# SADRŽAJ

<b>1. UVOD</b> .....	<b>1</b>
<b>1.1. Zadatak završnog rada</b> .....	<b>2</b>
<b>2. VREMENSKA INTERPOLACIJA SLIKE</b> .....	<b>3</b>
<b>2.1. Procjena pokreta u okviru povećanja vremenske rezolucije videosignala</b> .....	<b>4</b>
<b>2.2. Algoritmi za pronalazak odgovarajućeg bloka (Block-matching algorithm - BMA)</b> .....	<b>5</b>
2.2.1 Iscrpna pretraga (Exhaustive Search – ES).....	8
2.2.2 Pretraga u tri koraka (Three Step Search – TSS) .....	8
2.2.3 Nova pretraga u tri koraka (New Three Step Search - NTSS) .....	10
2.2.4 Dijamantna pretraga (Diamond Search - DS).....	11
<b>3. POVEĆANJE VREMENSKE REZOLUCIJE VIDEOSIGNALA UZ RAZLIČITE POSTUPKE PRETRAŽIVANJA SLIKE</b> .....	<b>14</b>
<b>3.1. Rezultati za postupak pretrage Exhaustive Search (ES)</b> .....	<b>18</b>
<b>3.2. Rezultati za postupak pretrage Diamond Search (DS)</b> .....	<b>23</b>
<b>3.3. Rezultati za postupak pretrage New Three Step Search (NTSS)</b> .....	<b>28</b>
<b>3.4. Usporedba rezultata</b> .....	<b>33</b>
<b>4. ZAKLJUČAK</b> .....	<b>40</b>
<b>POPIS LITERATURE</b> .....	<b>42</b>
<b>SAŽETAK</b> .....	<b>43</b>
<b>ABSTRACT</b> .....	<b>44</b>

## 1. UVOD

Videozapisi su danas vrlo popularan tip zabavnog sadržaja i prijenosa informacija. Razvojem tehnologije došli smo do toga da se i mobitelima mogu snimiti videozapisi visoke kvalitete što je samo pridonijelo njihovoj popularnosti. No to znači da svaki videozapis pa i onaj snimljen na mobitelu zauzima sve više i više prostora što ga čini nepogodnim za prijenos i pohranu. Primjerice, prijenos sadržaja putem interneta trajao bi duže vrijeme ili bi bio nemoguć ako bi se prenosio u punoj veličini. Isto vrijedi i za prikazivanje videozapisa u realnom vremenu, videopozive koje danas imamo na mnogim aplikacijama te videokonferencije koje se često koriste u poslovnom okruženju, a odnedavno sve više i u obrazovanju. Još jedan od bitnih razloga za smanjivanje veličine videozapisa je spremanje na medije za pohranu podataka. Zbog svega navedenog potrebno je smanjiti veličinu videozapisa, a to se može postići kompresijom u postupku kodiranja. Kodiranjem se naravno smanjuje kvaliteta videosignala, te se u nekim slučajevima kada se videozapis treba kodirati niskom stopom bitova zbog ograničene propusnosti prilikom prijenosa videozapisa mrežom, može dobiti veća kvaliteta videosignala na prijemu tako da se smanji vremenska rezolucija prije kodiranja, a potom se poveća poslije dekodiranja. Povećavanje vremenske rezolucije, to jest nadomještanje obrisanih sličica videozapisa čija je vremenska rezolucija smanjena povećat će njegovu kvaliteta, no kvaliteta videozapisa neće moći dostići kvalitetu originala. Razlog tomu je što više informacija bude obrisano prilikom kodiranja, teže će biti nadomjestiti te informacije te se može očekivati gora kvaliteta sličica koje će nadomjestiti originalne.

U fokusu ovog rada bit će procjena vektora pomaka u postupku povećanja vremenske rezolucije. Procjena pokreta je proces određivanja vektora pomaka koji opisuju transformaciju jedne 2D slike u drugu, najčešće susjednih sličica iz videozapisa. Vektori pomaka nam daju informacije o smjeru i pomaku objekata između dvije slike. Postoji više metoda koje se mogu koristiti za pronalazak vektora pomaka.

U radu će se prvo definirati i objasniti pojam vremenske interpolacije slike. Nakon toga naglasak će biti stavljen na načine procjene vektora pomaka, što je sastavni dio brojnih algoritama za povećanje vremenske rezolucije videosignala (vremenske interpolacije slike) te će biti analiziran utjecaj veličine bloka u postupku procjene vektora pomaka na kvalitetu interpolirane slike. U posljednjem poglavlju bit će prezentirani zaključci ovog rada.

## **1.1. Zadatak završnog rada**

Zadatak završnog rada je implementirati barem jedan od postojećih algoritama za procjenu pokreta te u njemu mijenjati veličinu bloka koji se koristi za procjenu vektora pomaka. Potrebno je ocijeniti kvalitetu interpoliranih slika dobivenih korištenjem različitih veličina blokova za procjenu vektora pomaka te zaključiti kakav je utjecaj veličine bloka prilikom procjene vektora pomaka na kvalitetu interpolirane slike.

## 2. VREMENSKA INTERPOLACIJA SLIKE

Vremenska rezolucija videa predstavlja broj sličica prikazanih u sekundi. Kao što je rečeno u uvodu, ponekad se prije samog kodiranja videozapisa smanji vremenska rezolucija istog tako da se izbace neke sličice. To omogućava lakši prijenos videozapisa, ali smanjuje mu kvalitetu. Zbog sve veće potrebe za velikom kvalitetom videosignala koji se prikazuje, često je potrebno u postupku dekodiranja povećati vremensku rezoluciju videosignala, što se ostvaruje vremenskom interpolacijom. Osim toga, kod nekih videozaslona kao što je LCD (eng. *Liquid-crystal display*), veća vremenska rezolucija videa smanjuje zamućenje kod pokreta (eng. *Motion bluriness*) i zato se i u takvim slučajevima zahtijeva efikasan algoritam za povećanje vremenske rezolucije. Algoritmi za povećanje vremenske rezolucije nazivaju se FRUC (eng. *Frame Rate Up-Conversion*) algoritmi. [1] FRUC algoritmi generiraju videozapis s većom vremenskom rezolucijom interpolacijom novih između postojećih slika u videozapisu manje vremenske rezolucije. Na slici 2.1. prikazan je općeniti postupak povećanja vremenske rezolucije videosignala.



Slika 2.1. Prikaz povećanja vremenske rezolucije

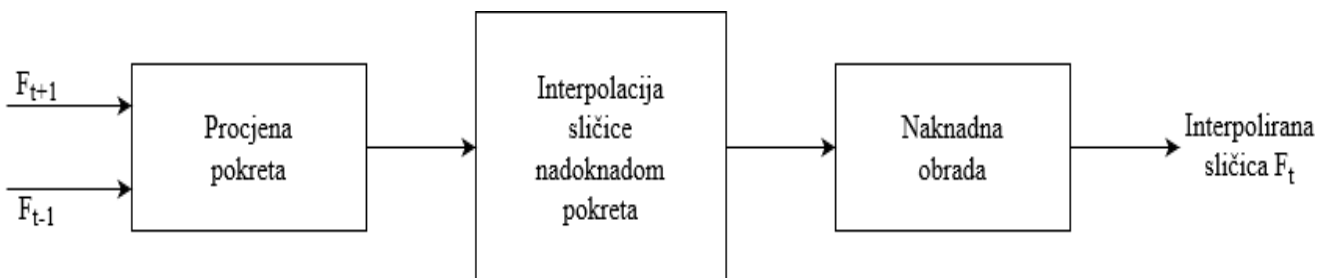
Na lijevoj strani slike 2.1. nalazi se niz slika s manjom vremenskom rezolucijom, a na desnoj strani nalazi se niz slika čija se vremenska rezolucija povećala. U desnom nizu slika sive sličice predstavljaju originalne sličice, a plave sličice predstavljaju interpolirane sličice koje su se dodale kako bi se povećala vremenska rezolucija. Ako su  $F_{t-1}$  i  $F_{t+1}$  dvije uzastopne slike u sekvenci s manjom vremenskom rezolucijom, tada se slika koja se interpolira između njih može dobiti korištenjem sljedećeg izraza:

$$F_t(x, y) = \alpha * F_{t-1}(x + (1 - \alpha) * v_x, y + (1 - \alpha) * v_y) + (1 - \alpha) * F_{t+1}(x - \alpha * v_x, y - \alpha * v_y) \quad (2-1)$$



gdje  $(x,y)$  predstavljaju lokaciju elemenata slike u sličicama,  $v = (v_x, v_y)$  predstavlja razliku u poziciji elemenata slike između  $F_{t-1}$  i  $F_{t+1}$  zbog pomaka, a  $\alpha$  predstavlja multiplikator koji prikazuje veličinu utjecaja sličica  $F_{t-1}$  i  $F_{t+1}$  na interpoliranu sličicu. [1]

FRUC metode mogu se podijeliti na one koje koriste procjenu pokreta i one koje ne koriste. FRUC metode koje ne koriste procjenu pokreta puno su jednostavnije, a dvije najčešće takve metode su metoda ponavljanja slike (eng. *FR, Frame Repetition*) i metoda usrednjavanja slike (eng. *FA, Frame Averaging*). Metoda ponavljanja slike opisana je jednadžbom (3-1) kada je vrijednost multiplikatora  $\alpha = 1$ . Ova metoda samo ponavlja prethodnu sličicu. Metoda usrednjavanja slike opisana je jednadžbom (3-1) s vrijednostima  $\alpha = 0.5$  i  $v_x = v_y = 0$ . Ona interpolira sliku računanjem srednje vrijednosti elemenata na istim lokacijama u dvjema uzastopnim sličicama. No metode koje ne koriste procjenu pokreta daju prihvatljive rezultate samo u videozapisima koji sadrže malu količinu pokreta jer mogu uzrokovati zamućenje i trzanje kod objekata koji se kreću. Složenije, uspješnije metode vremenske interpolacije sadrže procjenu pokreta, nadoknadu pokreta i naknadnu obradu interpolirane sličice i nazivaju se interpolacija sličice nadoknadom pokreta (eng. *MCFI, Motion-Compensated Frame Interpolation*). [1] Taj je proces prikazan na slici 2.2.:



Slika 2.2. Prikaz povećanja vremenske rezolucije MCFI algoritmom [1]

## 2.1. Procjena pokreta u okviru povećanja vremenske rezolucije videosignala

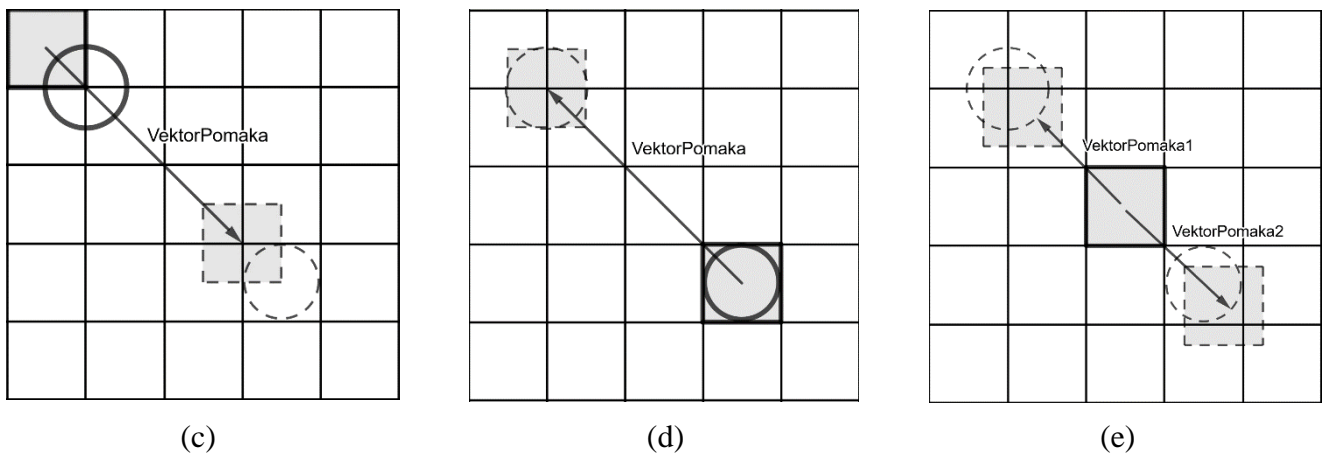
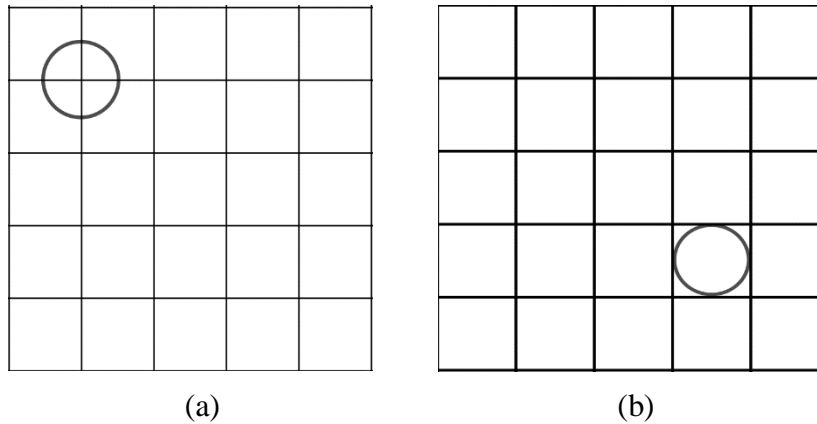
Procjena pokreta je proces određivanja vektora pomaka koji opisuju transformaciju jedne 2D slike u drugu, najčešće susjednih sličica iz nekog videozapisa. Procjena pokreta prvi je korak MCFI algoritma koji povećava vremensku rezoluciju videozapisa, a ujedno i najbitniji. Ona se koristi i prilikom kodiranja videozapisa, no vektori pomaka koji se dobiju prilikom procesa kodiranja najčešće nisu dovoljno precizni za MCFI algoritme jer oni ne prate kretanje objekata te se zbog toga rijetko koriste u tim algoritmima. Procjena pokreta ima vrlo bitan utjecaj na kvalitetu

interpolirane slike i zato je potrebno implementirati algoritam koji što preciznije određuje vektore pomaka.

Postoji više algoritama koji se koriste za svrhu procjene pokreta. U nekim se algoritmima koristi procjena pokreta temeljena na elementu slike koristeći optički tok, no ovakav algoritam zahtijeva puno računalnih resursa te je prezahtjevan za aplikacije u stvarnom vremenu. Zbog toga se najčešće koriste algoritmi za pronalazak odgovarajućeg bloka (eng. *Block-matching algorithm* – BMA) koji procjenjuju vektore pomaka na temelju pravokutnih blokova na koje se podijeli slika. Oni su puno jednostavniji za implementaciju. Procjena pokreta korištenjem BMA može biti jednosmjerna ili dvosmjerna s tim da se jednosmjerna može izvršiti unaprijed ili unazad. [2]

## **2.2. Algoritmi za pronalazak odgovarajućeg bloka (Block-matching algorithm - BMA)**

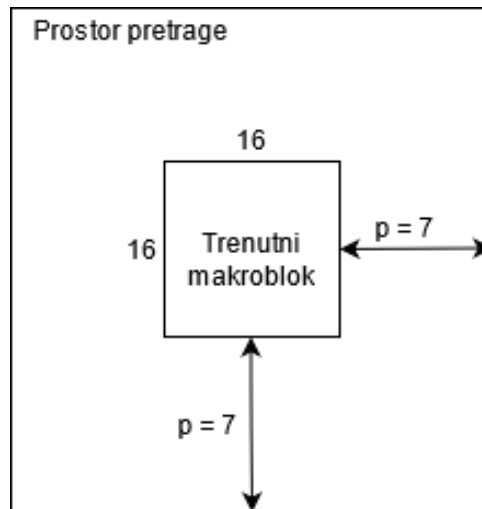
Kao što je rečeno u prošlom poglavlju, BMA se koristi za pronalazak najbližeg bloka u referentnoj slici za svrhu procjene pokreta. BMA može raditi jednosmjernu ili dvosmjernu procjenu. Jednosmjerna procjena može biti unaprijed ili unazad. Jednosmjerna procjena unaprijed radi tako da algoritam podijeli prethodnu sliku  $F_{t-1}$  na blokove i traži odgovarajući blok na sljedećoj slici  $F_{t+1}$ , dok jednosmjerna procjena unazad podijeli sljedeću sliku  $F_{t+1}$  na blokove i traži odgovarajući blok na prethodnoj slici  $F_{t-1}$ . Dvosmjerna procjena vrši procjenu pokreta tako da podijeli interpoliranu sliku  $F_t$  na blokove i istovremeno uspoređuje te blokove s blokovima na prethodnoj slici  $F_{t-1}$  i sljedećoj slici  $F_{t+1}$ . Razlika u poziciji između tih blokova predstavlja vektor pomaka koji opisuje kretanje objekta. Dok se u jednosmjernoj procjeni dobije samo jedan vektor pomaka, u dvosmjernoj procjeni se dobiju dva vektora pomaka istih veličina, ali suprotnih smjerova. Prednost dvosmjerne procjene je da ona ne stvara rupe ili preklapanja na interpoliranoj slici, ali nedostatak je da ne može napraviti preciznu procjenu za videozapise s velikim kretanjima. Kombinacija jednosmjerne procjene unaprijed i unazad daje više informacija o kretanju objekata i zato se one češće koriste. [1] Na slici 2.3. prikazana je prethodna i sljedeća slika u nizu te određivanje vektora pomaka jednosmjernom i dvosmjernom procjenom.



**Slika 2.3.** Prikaz jednosmjerne i dvosmjerne procjene pokreta: (a) Slika  $F_{t-1}$ , (b) Slika  $F_{t+1}$ , (c) Procjena unaprijed, (d) Procjena unazad, (e) Dvosmjerna procjena [1]

Slika za koju se žele odrediti vektori pomaka podijeli se u matricu blokova. Kao što je rečeno na početku ovog poglavlja, u jednosmjernoj procjeni to će biti ili prethodna ili sljedeća slika, ovisno o procjeni koja se radi, a u dvosmjernoj procjeni to je slika koja se interpolira  $F_t$ . Onda se pronalazi odgovarajući blok na prethodnoj ili sljedećoj slici ovisno o vrsti jednosmjerne procjene ili na obje slike ako se radi dvosmjerna procjena i kreira se vektor pomaka koji predstavlja razliku u poziciji između dva bloka. Kako bismo dobili što bolje podudaranje blokova, prostor pretrage se kreira tako da se na sve četiri strane bloka u referentnoj slici doda određeni broj elemenata slike. Broj koji određuje količinu elemenata slike koji čine prostor pretrage naziva se parametar pretrage  $p$ . Prostor pretrage se kreira i da bismo ograničili područje koje će se provjeravati jer najčešće nema potrebe provjeravati područje koje je daleko od trenutnog bloka, time je procjena pokreta brža i manje zahtjevnja. Za veće razlike u blokovima potreban je i veći parametar pretrage, no to dovodi do zahtjevnije i dugotrajnije procjene pokreta pa se parametar pretrage mora pažljivo birati. Kada je pronađen odgovarajući blok za sve blokove iz matrice, onda

je napravljena procjena pokreta za tu sliku. [3] Trenutni blok za koji se traži odgovarajući blok u referentnoj slici i prostor pretrage koji će se provjeravati prikazani su na slici 2.4.



**Slika 2.4.** Prikaz bloka veličine 16 x 16 elemenata slike i parametra pretrage veličine 7 elemenata slike [3]

Blok koji najviše odgovara trenutnom bloku određuje se pomoću jedne od sljedećih funkcija sličnosti: funkcija srednje apsolutne vrijednosti (eng. *Mean absolute difference – MAD*), funkcija srednje kvadratne pogreške (eng. *Mean squared error – MSE*), zbroj apsolutnih razlika (eng. *Sum of absolute differences – SAD*). Onaj blok koji ima najmanju vrijednost funkcije sličnosti je blok koji se najviše podudara s trenutnim. [2, 3] Funkcije su prikazane jednadžbama 2-2, 2-3 i 2-4:

$$MAD = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |C_{ij} - R_{ij}| \quad (2-2)$$

$$MSE = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (C_{ij} - R_{ij})^2 \quad (2-3)$$

gdje je  $N$  veličina stranice bloka, a  $C_{ij}$  i  $R_{ij}$  elementi slike koji se uspoređuju u trenutnom i referentnom bloku [3],

$$SAD = \sum_{x,y} |F_{t-1}(x, y) - F_{t+1}(x + MV_x, y + MV_y)| \quad (2-4)$$

gdje  $MV_x$  predstavlja horizontalnu komponentu vektora pomaka, a  $MV_y$  vertikalnu komponentu vektora pomaka. SAD se računa za elemente slika na istim pozicijama referentnog i trenutnog bloka. [2]

Postoji više različitih načina pretraživanja područja pretrage koji se koriste u BMA. Svaki se od njih razlikuje po složenosti, količini resursa i vremenu koje zahtijeva da izvrši pretragu, te

preciznosti pri određivanju vektora pomaka. Neki od najčešćih su: iscrpna pretraga (eng. *Exhaustive Search – ES*), pretraga u tri koraka (eng. *Three Step Search – TSS*), nova pretraga u tri koraka (eng. *New Three Step Search – NTSS*), jednostavna i efikasna pretraga (eng. *Simple and Efficient Search – SES*), pretraga uz prilagodljivo područje pretrage (eng. *Adaptive Rood Pattern Search – ARPS*), pretraga u četiri koraka (eng. *Four Step Search – FSS*), dijamantna pretraga (eng. *Diamond Search – DS*) i tako dalje. U nastavku rada detaljnije će biti opisana tri različita postupka pretraživanja te će isti biti implementirani prilikom procjene vektora pomaka. Isto tako će biti opisan i TSS način pretrage jer se na njemu temelji NTSS način pretrage koji će biti implementiran.

### **2.2.1 Iscrpna pretraga (Exhaustive Search – ES)**

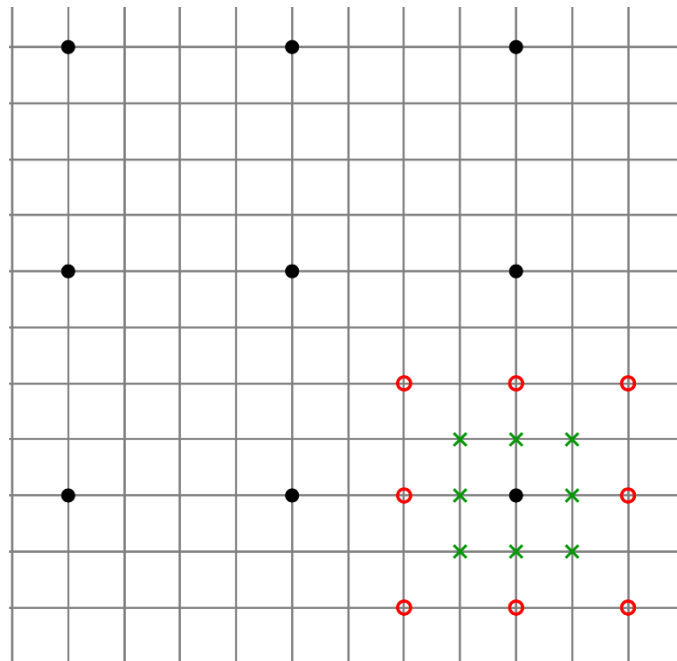
Prvi postupak koji će biti opisan zove se iscrpna pretraga. Ovaj se postupak nekad naziva i puna pretraga (eng. *Full Search – FS*), i on je računski najzahtjevniji od svih načina pretraživanja područja pretrage. Ovaj postupak pretrage uspoređuje trenutni blok sa svim mogućim blokovima u referentnoj slici. Zato uvijek pronalazi najbolje moguće podudaranje i zbog toga slike interpolirane ovim postupkom pretraživanja odgovarajućeg bloka imaju najveću kvalitetu. Drugi postupci pretraživanja pokušavaju imati što bolju kvalitetu interpolirane slike, to jest pokušavaju pronaći što bolje podudaranje blokova, ali uz što manje računanja, što znači da su brži od ovog, no nisu uvijek tako točni. Još jedan nedostatak ES postupka je što je veći prostor pretrage, broj računanja se povećava i sve je sporiji. [3]

ES radi na jednostavan način. Svaki blok iz trenutne slike uspoređuje sa svakim blokom u prostoru pretrage iz referentne slike. Kako bi odredio blok koji se najbolje podudara, računa se funkcija MAD za sve blokove. Odabire se onaj blok koji ima najmanju vrijednost te funkcije i postavlja se vektor pomaka kao razlika u pozicijama trenutnog i odabranog bloka.

### **2.2.2 Pretraga u tri koraka (Three Step Search – TSS)**

Pretraga u tri koraka jedan je od najbržih načina pretraživanja područja pretrage. Bio je jako popularan zbog svoje jednostavnosti i dobrih performansa. Ovaj postupak pretrage radi na sljedeći način. Postavi se veličina koraka koja se najčešće označava sa  $S$  na 4. Postavlja se 8 blokova koji se nalaze na udaljenosti  $S$  elemenata slike od bloka koji se postavi na koordinate (0, 0), a nalazi se na istom mjestu kao i blok na trenutnoj slici za koji se traži odgovarajući blok; ova pozicija bloka još se naziva i centar pretrage. Računa se vrijednost funkcije MAD svih 9 blokova, te se odabire onaj koji ima najmanju vrijednost. Taj se blok postavi kao novi centar pretrage i veličina koraka  $S$  se prepolovi. Ova se pretraga ponavlja sve dok veličina koraka  $S$  nije 1. Na kraju,

blok koji daje najmanju vrijednost MAD funkcije od svih pretraženih blokova je taj koji se najbolje podudara s trenutnim i kreira se vektor pomaka kao razlika u poziciji trenutnog i odabranog bloka. [4] Za parametar pretrage  $p = 7$ , ES će izračunati vrijednost za 255 blokova, dok TSS računa vrijednost za 25 blokova. [3] Vidimo da TSS značajno smanjuje broj blokova koji će se provjeriti, a time i vrijeme potrebno da se procjena pokreta izvrši. Prikaz rada ovoga načina pretrage je na slici 2.5.

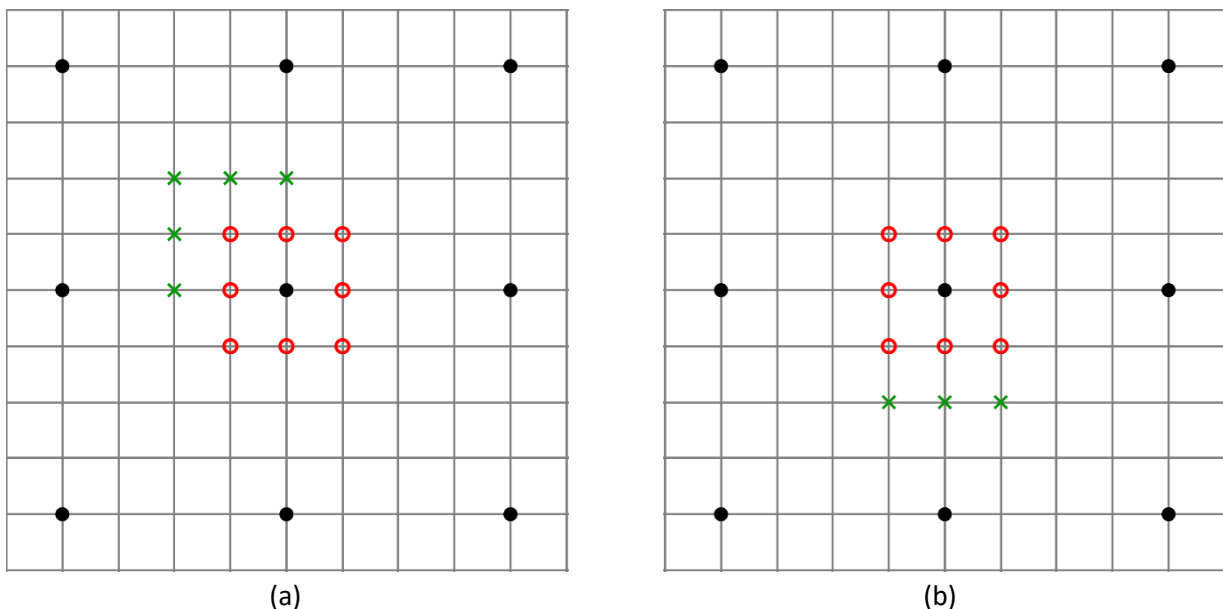


*Slika 2.5. Princip rada TSS postupka pretrage [4]*

Na slici 2.5. vidimo kako radi TSS postupak pretrage. Početna veličina koraka  $S$  je 4. Prvih 9 pozicija blokova koji se provjeravaju označeni su crnim punim točkama. Onaj blok koji ima najmanju vrijednost funkcije MAD je novi centar pretrage, a to je blok na koordinatama (4, -4). Veličina koraka smanji se za pola, sada je 2. Drugih 8 pozicija blokova koji se provjeravaju označeni su crvenim praznim točkama. Opet se provjeravaju vrijednosti funkcije MAD za sve blokove, najmanji se u ovom slučaju nalazi na (4, -6) i on je novi centar pretrage. Veličina koraka sada je 1, i to znači da je gotova pretraga. Zadnjih 8 pozicija blokova koji se provjeravaju označeni su zelenim križićima. Odabran je onaj blok koji ima najmanju vrijednost funkcije MAD. Postavlja se vektor pomaka kao razlika u pozicijama trenutnog i odabranog bloka.

### 2.2.3 Nova pretraga u tri koraka (New Three Step Search - NTSS)

NTSS se razlikuje od TSS postupka pretrage jer može završiti prije nego što provjeri sve blokove; time se smanjuje proces računanja i ubrzava procjena pokreta. Ovo je jedan od prvih široko korištenih brzih načina pretraživanja i bio je implementiran u ranijim standardima videokodiranja kao što su MPEG 1 i H.261. U prvom koraku ima 8 dodatnih blokova koje provjerava uz 8 koji su se nalazili u TSS-u. Ti novi blokovi nalaze se oko centra pretrage na udaljenosti od 1 elementa slike. Centar pretrage je onaj blok u referentnoj slici koji se nalazi na istom mjestu kao i blok za koji se traži vektor pomaka u trenutnoj slici. Izračunava se vrijednost funkcije MAD svih početnih 17 blokova. Ako se najmanja vrijednost te funkcije nalazi na centru pretrage, pretraga se prekida i vektor pomaka je na (0,0). Ako se najmanja vrijednost nalazi na jednom od 8 novih dodanih blokova, centar pretrage se postavlja na taj blok i izračunavaju se vrijednosti funkcije MAD njegovih susjeda. Ovisno o bloku na koji smo postavili centar pretrage on može imati 3 ili 5 susjednih blokova čije će se vrijednosti morati računati. Nakon toga kreira se vektor pomaka i pretraga se prekida. Ako je najmanja vrijednost na jednom od blokova koji su se nalazili i u TSS postupku pretrage, nastavlja se normalna TSS procedura koja je opisana ranije u radu. U najgorem slučaju NTSS će provjeriti 33 blokova, dok TSS provjerava 25. No zbog mogućnosti ranijeg prekida pretrage, NTSS može provjeriti u najboljem slučaju 17 blokova. [5] Prikaz rada ove pretrage je na slici 2.6.



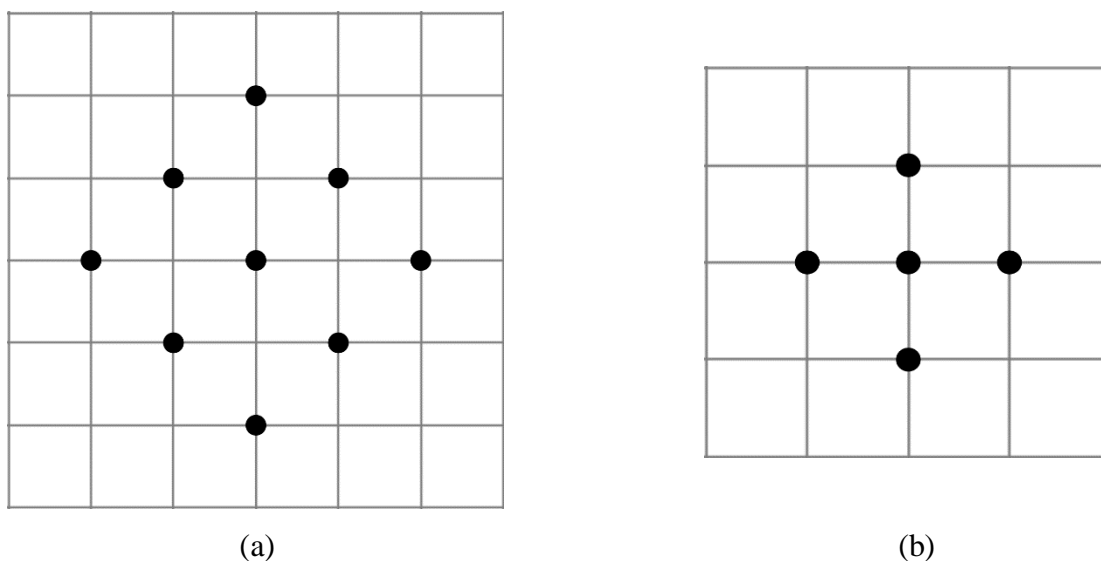
*Slika 2.6. Princip rada NTSS postupka pretrage: (a) 1. slučaj, (b) 2. slučaj [5]*

Na slici 2.6. vidimo prikaz kako radi NTSS postupak pretrage. Pune crne točke označavaju pozicije blokova koje smo imali i u TSS-u, a prazne crvene točke 8 novih dodatnih pozicija blokova. U prvom slučaju slike 2.6. vidimo primjer kada postupak pretrage pretraži 22 bloka, što je za 3 manje od TSS-a. Prvo je pretraženo 17 početnih blokova. Najmanju vrijednost imao je blok na poziciji (-1,1). Onda pretražuje novih 5 blokova čije su pozicije označene kao zeleni križići, jer su mu 3 susjeda već pretražena i to je ukupno 22 pretražena bloka.

U drugom slučaju slike 2.6. je primjer kada NTSS pretraži 20 blokova, što je za 5 manje od TSS-a. Ovdje je nakon početne pretrage najmanju vrijednost imao blok na poziciji (0, -1). Onda je pretražio nova 3 bloka čije su pozicije označene kao zeleni križići jer su mu vrijednosti ostalih susjeda već izračunate. Može se zaključiti da ako najmanju vrijednost ima blok čija je pozicija jedna od 4 praznih crvenih točaka koja se nalazi horizontalno ili vertikalno od centra pretrage NTSS ukupno pretraži 20 blokova. Ako najmanju vrijednost ima blok čija je pozicija jedna od 4 praznih crvenih točaka koja se nalazi dijagonalno od centra pretrage, onda NTSS ukupno pretraži 22 bloka.

#### 2.2.4 Dijamantna pretraga (Diamond Search - DS)

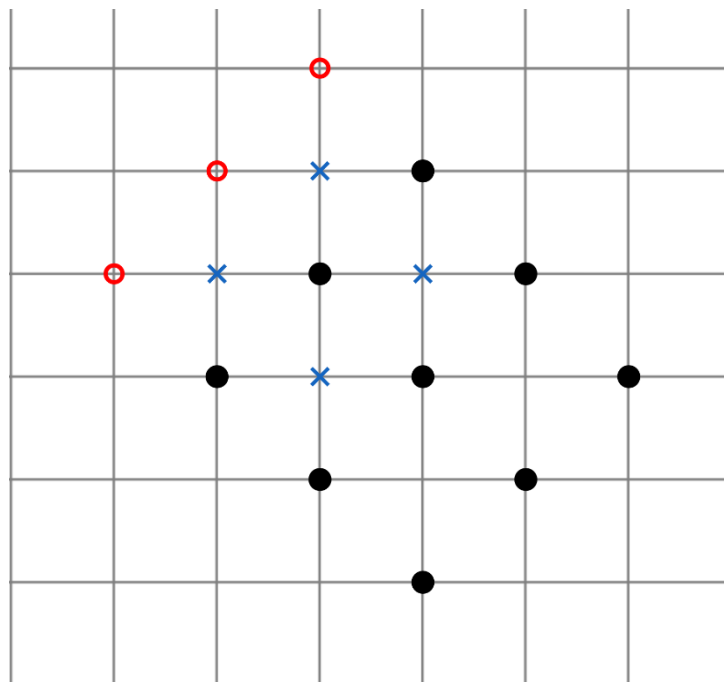
DS postupak pretrage sastoji se od dva obrasca pretrage. Prvi se zove veliki dijamantni obrazac pretrage (eng. *Large diamond search pattern, LDSP*), a drugi mali dijamantni obrazac pretrage (eng. *Small diamond search pattern, SDSP*). LDSP se sastoji od 9 blokova koji se provjeravaju gdje 8 blokova okružuju centralni kako bi napravili oblik dijamanta, a SDSP od 5 blokova koji prave manji oblik dijamanta. [6] LDSP obrazac prikazan je pod (a) na slici 2.7., a SDSP pod (b).



Slika 2.7. Dva obrasca pretrage: (a) LDSP, (b) SDSP [6]



Ovaj postupak pretrage radi na sljedeći način. Pretraga započinje na centru s LDSP obrascem. Centar pretrage predstavlja poziciju bloka na referentnoj slici koji ima iste koordinate kao i blok za koji se traži vektor pomaka na trenutnoj slici. Provjerava se svih 9 blokova, ako minimalnu vrijednost MAD funkcije ima blok u centru, onda se prelazi na SDSP obrazac, a ako ima jedan od ostalih 8 blokova, onda se postavlja centar pretrage na taj blok i ponovno pokreće LDSP. LDSP se ponavlja sve dok minimalnu vrijednost MAD funkcije ne bude imao centar pretrage. Kada se pređe na SDSP obrazac, provjerava se svih 5 blokova i onaj koji ima najmanju vrijednost je blok koji najbolje odgovara trenutnom. [6] Prikaz rada ovog postupka je na slici 2.8.



*Slika 2.8. Prikaza rada DS postupka pretrage [6]*

Crne pune točke predstavljaju pozicije blokova početne LDSP pretrage oko centra (0,0). Minimalnu vrijednost MAD funkcije imao je blok na poziciji (-1,1). Crvene prazne točke predstavljaju nove pozicije blokova koji se moraju pretražiti jer se centar pretrage premjestio na (-1,1). Minimalnu vrijednost je opet imao blok na poziciji (-1,1), no ovaj puta on je bio centar pretrage te se prelazi na SDSP pretragu. Plavi križići predstavljaju nove pozicije blokova koji se moraju pretražiti uz centar pretrage. Jedan od tih 5 blokova će se najviše podudarati s trenutnim blokom i kreirat će se vektor pomaka kao razlika u poziciji trenutnog i tog bloka.

DS postupak pretrage nema ograničenja za količinu koraka koje može napraviti dok ne pronade minimum, prostor pretrage mu nije ni premalen ni prevelik te zbog toga pronalazi minimalnu vrijednost odgovarajućeg bloka jako precizno. Kada se koristi DS postupak pretrage, interpolirane

slike su po kvaliteti jako slične interpoliranim slikama kada se koristi ES postupak pretrage ali uz puno brže vrijeme izvođenja. [3] Isto tako, simulacije prema [6] su prikazale da ovaj postupak ima puno bolje performanse od TSS postupka pretrage, te da postiže slične vrijednosti kao i NTSS postupak, ali uz otprilike 22% manje računanja.

### 3. POVEĆANJE VREMENSKE REZOLUCIJE VIDEOSIGNALA UZ RAZLIČITE POSTUPKE PRETRAŽIVANJA SLIKE

Kao što je rečeno u uvodu, cilj ovog završnog rada je implementirati neku metodu za procjenu pokreta, i na toj metodi testirati utjecaj veličine bloka na kvalitetu interpolirane slike. U ovom radu implementirani će biti *Exhaustive Search* (ES), *New Three Step Search* (NTSS) i *Diamond Search* (DS) postupci pretrage. Vrsta procjene pokreta koja će se koristiti je dvosmjerna. Zadatak će biti izvršen u programskom paketu FFmpeg. FFmpeg je besplatni softver otvorenog koda koji se sastoji od mnogo programa i biblioteka za rad s videozapisima, audiozapisima i ostalim multimedijским datotekama. On sadrži naredbu za smanjivanje vremenske rezolucije videozapisa i naredbu koja omogućava povećanje vremenske rezolucije videozapisa interpolacijom sličica. Kvaliteta interpolirane slike bit će ocijenjena PSNR (eng. *peak signal-to-noise ratio*) metodom, koja se također može izračunati u FFmpeg-u. PSNR je omjer maksimalne vrijednosti signala sa šumom koji utječe na taj signal. Ova se metoda često koristi kao mjera kvalitete između originalne i interpolirane slike. Što je veći PSNR, to je kvaliteta interpolirane slike veća i ona je sličnija originalnoj. Prednost PSNR je da je brz i jednostavan za implementaciju za razliku od nekih drugih metoda. Nedostatak je da se on strogo temelji na brojčanim usporedbama, a ne uzima u obzir sadržaj slike. PSNR se računa prema formuli 3-1 [7]:

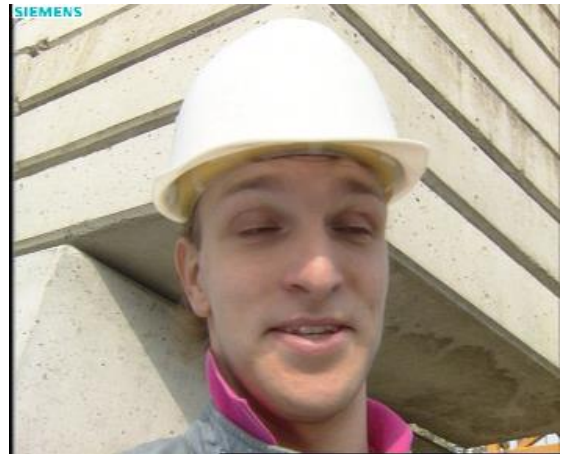
$$PSNR = 20 \log_{10} \left( \frac{MAX_f}{\sqrt{MSE}} \right) \quad (3-1)$$

Gdje  $MAX_f$  predstavlja maksimalnu vrijednost signala koji se nalazi na originalnoj slici, a  $MSE$  je vrijednost ranije spomenute funkcije za pretragu odgovarajućeg bloka. Mjerna jedinica PSNR vrijednosti je decibel [dB].

Utjecaj veličine bloka bit će analiziran za pet različitih videosekvenci: *Akiyo*, *Foreman*, *Coastguard*, *Container*, *Stefan*. Sve su sekvence preuzete sa [8, 9] gdje se nalaze videosekvence za testiranje u 4:2:0 YUV formatu. Prostorna rezolucija svih sekvenci je 352 x 288 elemenata slike, a vremenska rezolucija je 25 slika u sekundi. Sve videosekvence sadrže ukupno 300 slika. Prva slika svake sekvence korištene za analizu prikazana je na slici 3.1.



(a)



(b)



(c)



(d)



(e)

*Slika 3.1. Prva sličica svih videosekvenci korištenih za analizu: (a) Akiyo, (b) Foreman, (c) Coastguard (d) Container (e) Stefan*

Sekvenca *Akiyo* sastoji se od voditeljice koja čita vijesti, videozapis je dosta miran, kamera se ne pomiče te jedini pokret čini voditeljica i ti pokreti su mali. Sekvenca *Foreman* prikazuje graditelja, prvo je prikazano njegovo lice dok priča u kameru, kamera nije mirna jer je graditelj pomiče prilikom pričanja te je na kraju okreće i prikazuje gradilište. Pomiče se samo jedan objekt i to je graditelj, kada se kamera okrene pri kraju više nema objekata koji se kreću. Sekvenca *Coastguard* prikazuje prvo manji gliser kako se približava većoj brodicu, oni se mimoilaze i kamera onda ostaje pratiti samo brodicu. Ovdje imamo pokrete više objekata i u jednom trenutku nagli pomak kamere. Sekvenca *Container* sadrži nepokretnu kameru te dva broda koja se kreću u istom smjeru. Sekvenca *Stefan* je videosekvenca s najviše pokreta. Kamera je konstantno u pokretu, ona prati igrača tenisa koji je također stalno u pokretu. I kamera i tenisač u određenim trenucima prave jako brze pokrete. Jedini drugi objekt u sekvenci koji se kreće je loptica. Analiza za svih pet sekvenci bit će izvršena za svih 300 sličica.

Slijedi opis izvršenja zadatka rada i naredbi koje su korištene u FFmpegu. FFmpeg se koristi pomoću komandne linije. Prvo se u komandnoj liniji dođe do mape koja sadrži videozapis s kojim želimo raditi i onda koristimo naredbe koje su dio FFmpega. Svaka naredba u komandnoj liniji počinje sa *ffmpeg*. Videosekvence će prvo biti konvertirane iz YUV formata u mp4. Primjer naredbe je sljedeći:

```
ffmpeg -f rawvideo -vcodec rawvideo -s 352x288 -r 25 -pix_fmt yuv420p -i input_file.yuv -c:v libx264 -preset ultrafast -qp 0 output_file.mp4
```

Gdje *-f rawvideo* predstavlja da ulazni format nije procesiran, to jest da se radi o izvornom spremniku videa koji ne sadrži potrebne parametre videozapisa te ih korisnik sam mora navesti, s parametrom *-vcodec rawvideo* se postavlja da podatci koji se nalaze u izvornom spremniku videa nisu komprimirani, *-s 352x288* predstavlja prostornu rezoluciju videozapisa, *-r 25* predstavlja vremensku rezoluciju, *-pix\_fmt yuv420p* da je odabran YUV4:2:0 format poduzorkovanja U i V komponenti, *-i input\_file.yuv* predstavlja ulaznu yuv datoteku, *-c:v libx264* nam govori da je za kodiranje izabrana libx264 biblioteka koja sadrži H.264 format kodiranja, *-preset ultrafast* predstavlja brzinu kompresije, *-qp 0* da se želi postići maksimalna kvaliteta, i na kraju *output\_file.mp4* izlaznu datoteku u mp4 formatu. Nakon toga će biti smanjena vremenska rezolucija svih videosekvenci za pola, to jest sa 25 slika u sekundi na 12.5. Primjer naredbe smanjivanja vremenske rezolucije:

```
ffmpeg -i input_file.mp4 -filter:v fps=12.5 lowOutput_file.mp4
```

Gdje *-i input\_file.mp4* predstavlja ulaznu datoteku, *-filter:v fps=12.5* filter koji smanjuje vremensku rezolucije videozapisa na 12.5 te *lowOutput\_file.mp4* izlaznu datoteku sa smanjenom vremenskom rezolucijom. Potom će se MCFI algoritmom povećati vremenska rezolucija nazad na originalnu. U ovome koraku će se mijenjati veličina bloka i vrsta procjene pokreta. Primjer naredbe:

```
ffmpeg -i lowInput_file.mp4 -filter  
"minterpolate='mi_mode=mci:me_mode=bidir:me=esa:mb_size=16:search_param=32:fp  
s=25:'" interpolatedOutput_file.mp4
```

Gdje *-i lowInput\_file.mp4* predstavlja ulaznu datoteku sa smanjenom vremenskom rezolucijom, *-filter minterpolate* predstavlja naredbu za povećanje vremenske rezolucije i ona se sastoji od sljedećih parametara: *mi\_mode* predstavlja vrstu vremenske interpolacije, tu se mogu odabrati dvije vremenske interpolacije koje ne koriste procjenu pokreta, a to su metoda ponavljanja slike (*dup*) i metoda usrednjavanja slike (*blend*). Treća opcija je da se odabere vremenska interpolacija koja koristi procjenu pokreta MCFI (*mci*), to je opcija koja se mora odabrati za izvršenje ovog rada. Ako se odabere MCFI, mogu se birati sljedeći parametri: parametar *me\_mode* predstavlja vrstu procjene pokreta, mogu se izabrati dvije vrste, a to su dvosmjerna procjena pokreta (*bidir*) i bilateralna procjena pokreta (*bilat*). Izabrana je dvosmjerna procjena pokreta koja je teorijski opisana u radu. Parametar *me* predstavlja način pretraživanja područja pretrage koji će se koristiti za procjenu pokreta, tri implementirana postupka pretrage izabiru se sljedećim vrijednostima: *exhaustive search* sa *esa*, *diamond search* sa *ds*, *new three step search* sa *ntss*. Parametar *mb\_size* predstavlja veličinu bloka, parametar *search\_param* je veličina parametra pretrage *p*, parametar *fps* je iznos sličica u sekundi za izlaznu datoteku, tj. na koliko se želi povećati vremenska rezolucija te *interpolatedOutput\_file.mp4* je izlazna datoteka čija je vremenska rezolucija povećana interpolacijom s navedenim parametrima. Veličine bloka koje se testiraju su sljedeće: 16 x 16 elemenata slike, 8 x 8 elemenata slike, 4 x 4 elemenata slike. Parametar pretrage *p* je stalno postavljen na 32. Na kraju će se izračunati prosječni PSNR svake sličice usporedbom originalnog videozapisa i videozapisa čija je vremenska rezolucija povećana interpolacijom. Naredba za izračun PSNR vrijednosti je sljedeća:

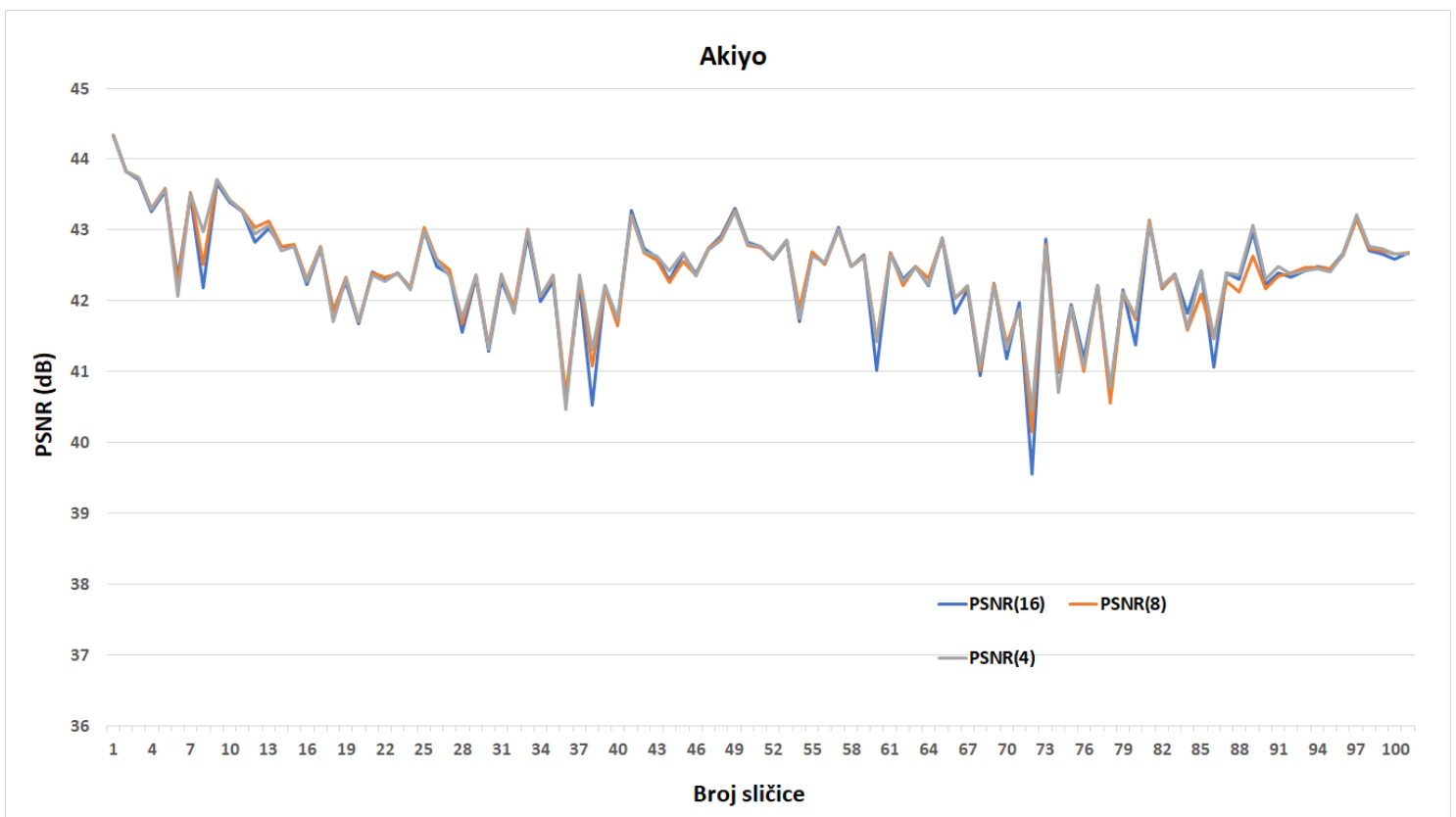
```
ffmpeg -i videoToCompare.mp4 -i originalVideo.mp4 -lavfi  
psnr=stats_file=psnr_logfile.txt -f null -
```

Gdje *-i videoToCompare.mp4* predstavlja videozapis čija je vremenska rezolucija povećana interpolacijom na istu vremensku rezoluciju kakvu ima i originalni videozapis, *-i originalVideo.mp4* predstavlja originalni videozapis, *-lavfi* predstavlja biblioteku koja sadrži filter

za računanje PSNR vrijednosti,  $psnr=stats\_file=psnr\_logfile.txt$  predstavlja izračun PSNR vrijednosti i zapis tih vrijednosti za svaku sličicu u tekstualnu datoteku. U tekstualnoj datoteci su zapisane broj sličice, prosječna vrijednost MSE funkcije i vrijednost MSE funkcije svake YUV komponente za svaku sličicu, prosječna PSNR vrijednost te PSNR vrijednost svake YUV komponente za svaku sličicu. Prosječna PSNR vrijednost za svaku sličicu je prikazana na grafovima koji slijede u rezultatima, a prosječna PSNR vrijednost svih sličica je izračunata u Microsoft Excelu pomoću funkcije *Average* te prikazana tablicama u poglavlju 3.4. Usporedit će se dobivene PSNR vrijednosti i proučiti utjecaj bloka na njegovu vrijednost.

### 3.1. Rezultati za postupak pretrage Exhaustive Search (ES)

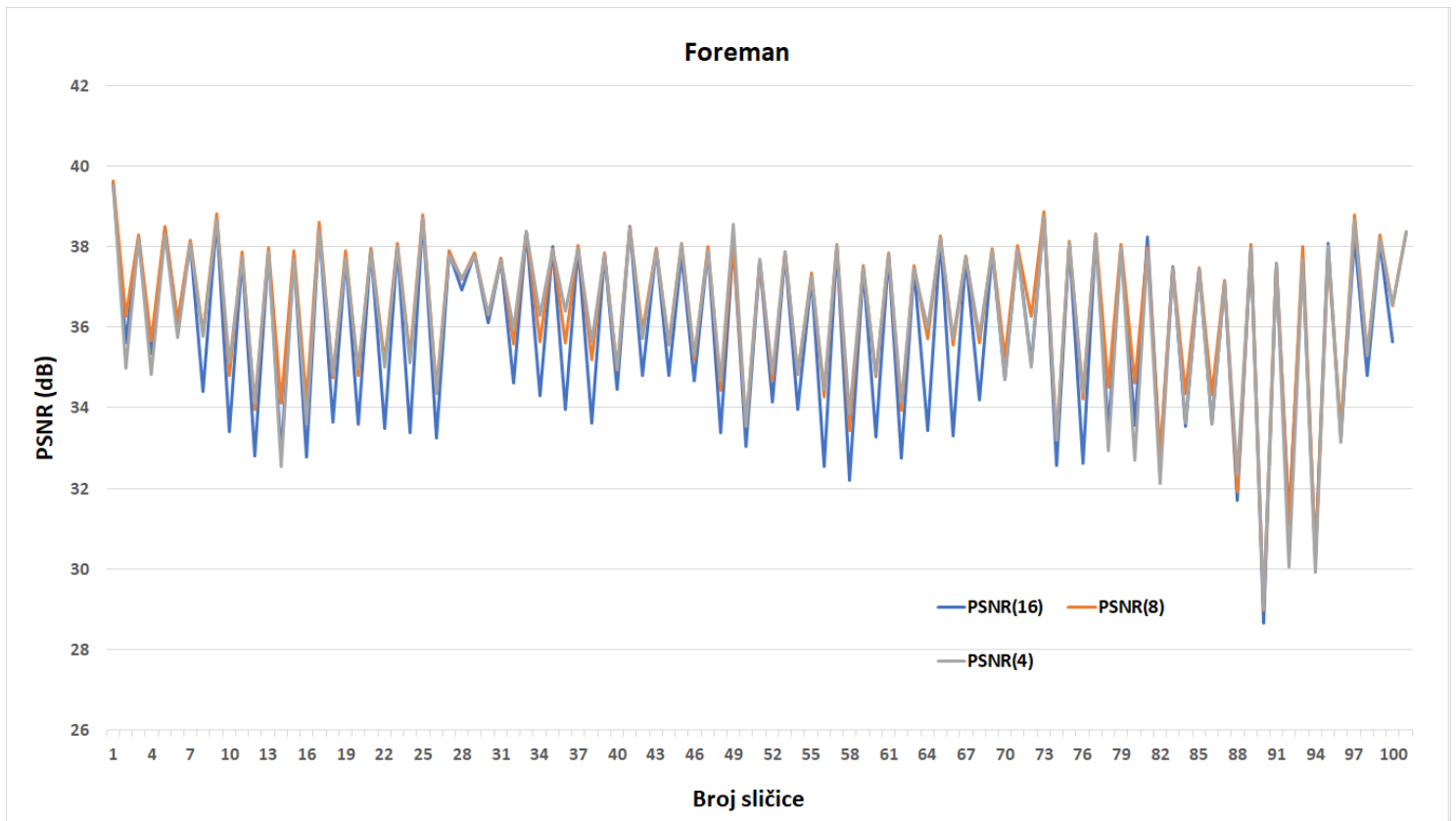
Sada će biti prikazani rezultati PSNR vrijednosti koji su dobiveni uspoređivanjem originalnog videa i videa čija je vremenska rezolucija povećana korištenjem ES postupka pretraživanja za svih pet videosekvenci. Na jednom grafu bit će prikazane prosječne PSNR vrijednosti sličica jedne videosekvence za sve tri testne veličine bloka. Prikazat će se vrijednosti za prvih 100 sličica kako bi graf bio jasniji. Plava linija predstavlja vrijednost PSNR za veličinu bloka 16 x 16 elemenata slike, narančasta za veličinu bloka 8 x 8 elemenata slike, a siva za veličinu bloka 4 x 4 elemenata slike. Prvo je prikazana videosequencia *Akiyo*:



Slika 3.2. Prosječne PSNR vrijednosti sličica videosekvence *Akiyo*

Ova videosekvencija je vrlo mirna, nema veliku količinu pokreta. Zbog toga, procjena pokreta je vrlo precizna bez obzira o kojoj veličini bloka se radi i rezultati su vrlo slični pa se može reći da veličina bloka u ovom slučaju nema veliki utjecaj na kvalitetu interpoliranih sličica. Procjena pokreta također ima veliki utjecaj na kvalitetu interpoliranih sličica, a s obzirom da je ovdje procjena pokreta bila precizna to se očitivalo u vrlo visokim PSNR vrijednostima kroz cijeli graf.

Sekvenca *Foreman*:



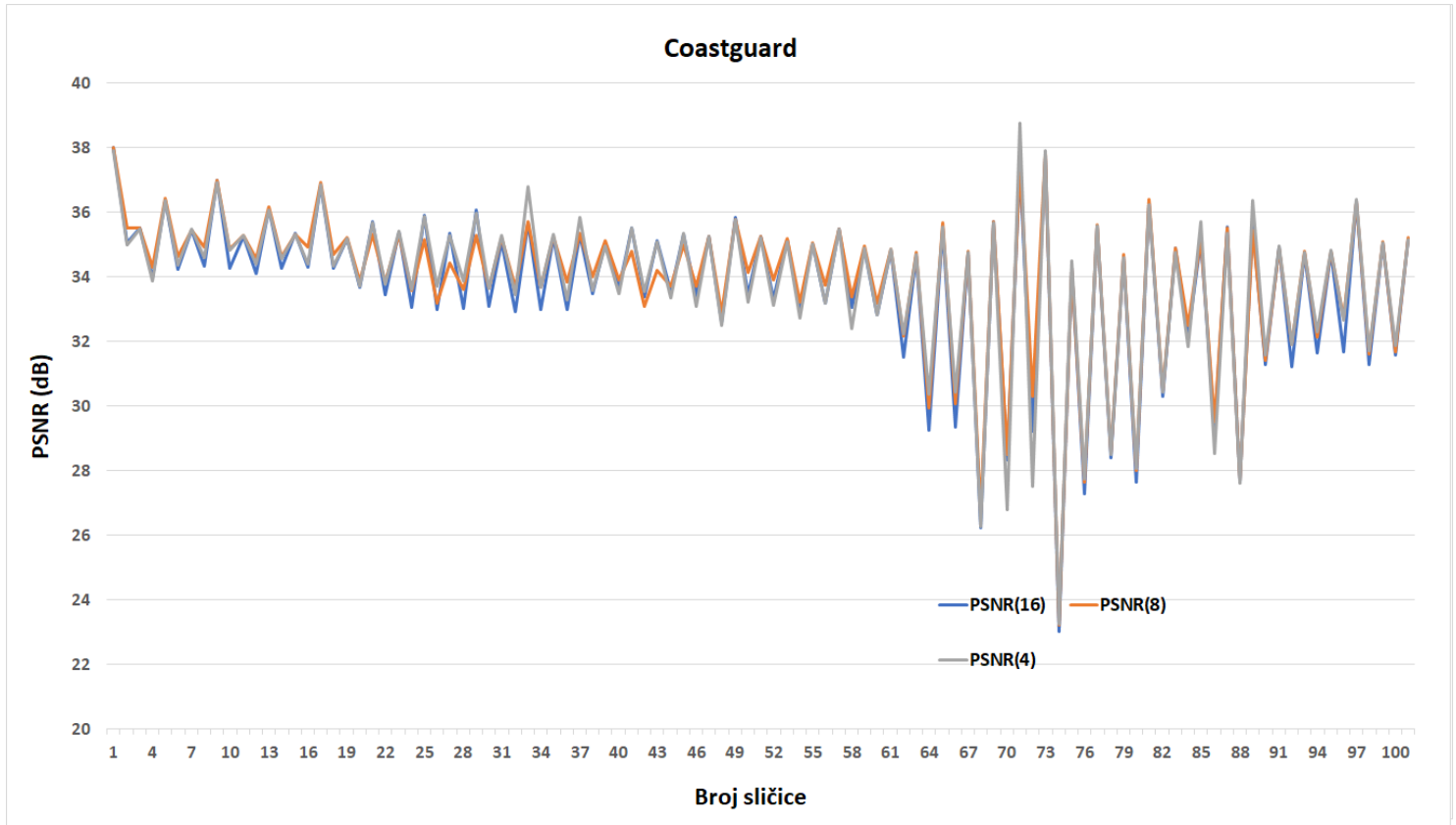
Slika 3.3. Prosječne PSNR vrijednosti sličica videosekvence *Foreman*

Za videosekvencu *Foreman* vrijednost PSNR bila jako promjenjiva, padala je i rasla između 38 dB i 33 dB skoro konstantno, a u jednom trenutku je pala i ispod 30 dB. Razlog manjim PSNR vrijednostima i stalnom padu i rastu grafa je da ova sekvenca ima veće pomake objekta i kamera se konstantno pomiče. Zbog toga je veća vjerojatnost da procjena pokreta neće biti toliko precizna, a ona ima jako veliki utjecaj na kvalitetu interpolirane slike. Zbog lošije procjene pokreta, utjecaj bloka se ovdje nešto bolje vidi nego u videosekvenci *Akiyo*. Može se vidjeti na grafu da je PSNR vrijednost kada se koristila veličina bloka 16 x 16 elemenata slike često bila najmanja za prvih 100 sličica. Zbog toga, možemo pretpostaviti da u prvih 100 sličica ima puno mali kretnji



koje veća veličina bloka kao 16 x 16 elemenata slike u ES postupku pretrage ne može najbolje procijeniti.

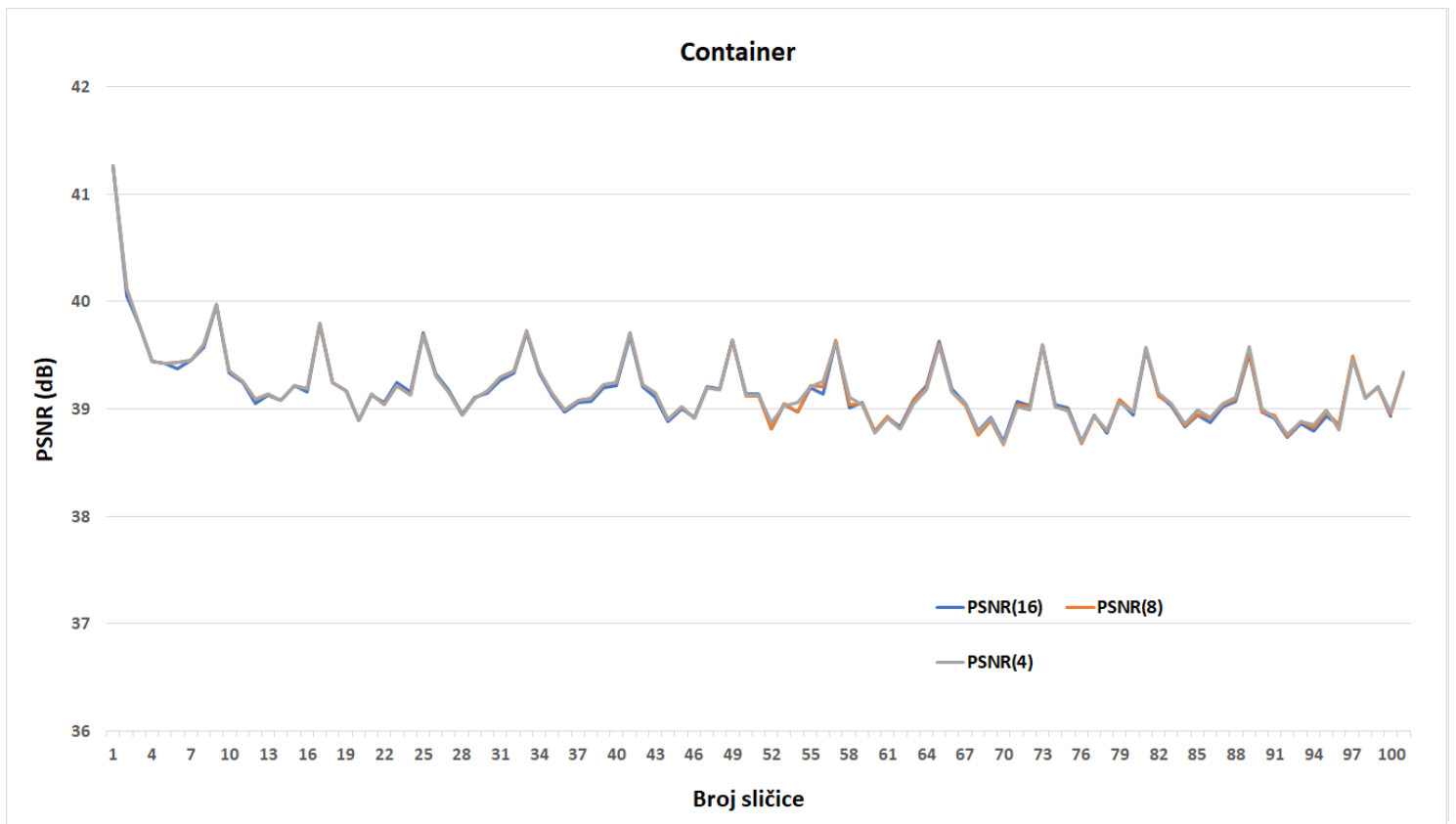
Sekvenca *Coastguard*:



Slika 3.4. Prosječne PSNR vrijednosti sličica videosekvence *Coastguard*

Kod videosekvence *Coastguard* PSNR vrijednost je bila konstantnija na prvom dijelu grafa, kretala se između 33 i 37 dB. Potom oko 60. sličice kreću veliki porasti i padovi kvalitete, a tu je u videosekvenci bio veliki pomak kamere. Najveći je pad kvalitete oko sličice 73 gdje je PSNR vrijednost došla do otprilike 23dB dok je još trajao taj veliki pomak kamere. Zbog tog velikog pomaka kamere procjena pokreta nije toliko precizna, a ona ima veliki utjecaj na kvalitetu interpolirane slike i to je rezultiralo tim velikim padovima. Utjecaj veličine bloka nije toliko velik, no vidi se na grafu da najniže PSNR vrijednosti najčešće budu kada se koristi veličina bloka 16 x 16 elemenata slike ili 4 x 4 elemenata slike. To bi značilo da je veličina bloka 8 x 8 elemenata slike dobar kompromis za male i velike pomake ove sekvence u ovom postupku pretrage.

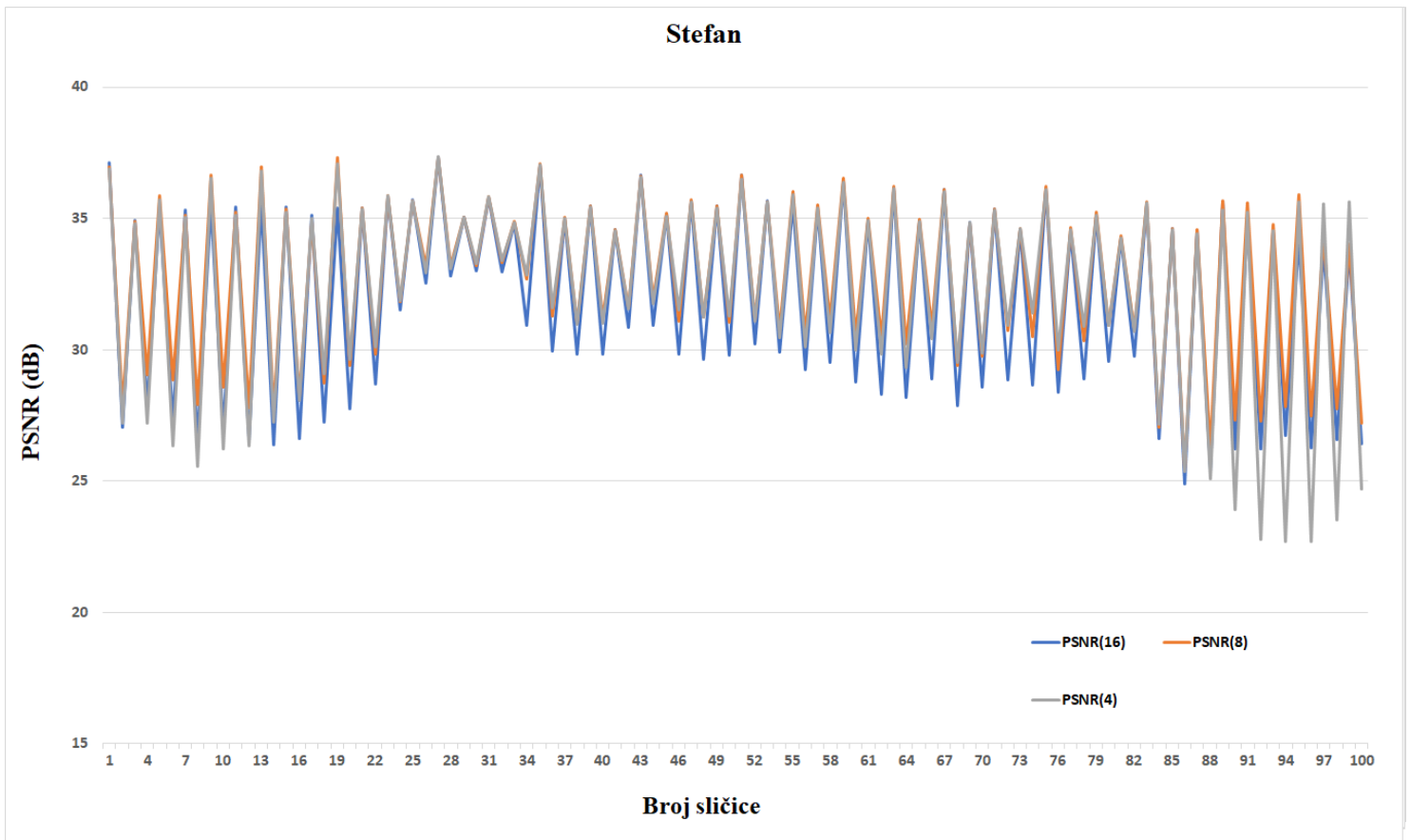
Sekvenca *Container*:



*Slika 3.5. Prosječne PSNR vrijednosti sličica videosekvenca Container*

Videosekvenca *Container* je najmirnija sekvenca dosad, nema pomaka kamere te pokreti objekata su mali. Zbog toga, procjena pokreta može jako precizno odrediti vektore pomaka, te je kvaliteta interpoliranih sličica najbližnja dosad pa je i na grafu teško pronaći razliku između PSNR vrijednosti. PSNR vrijednosti su stalno jako slične, sa visokim vrijednostima između 39 i 40 dB, bez velikih padova kakav je bio slučaj u sekvencama sa više pokreta. Po grafu možemo zaključiti da je veličina bloka 4 x 4 elemenata slike najčešće dala najbolje PSNR vrijednosti jer je siva linija za prvih 100 sličica većinom na vrhu.

Posljednja sekvenca *Stefan*:

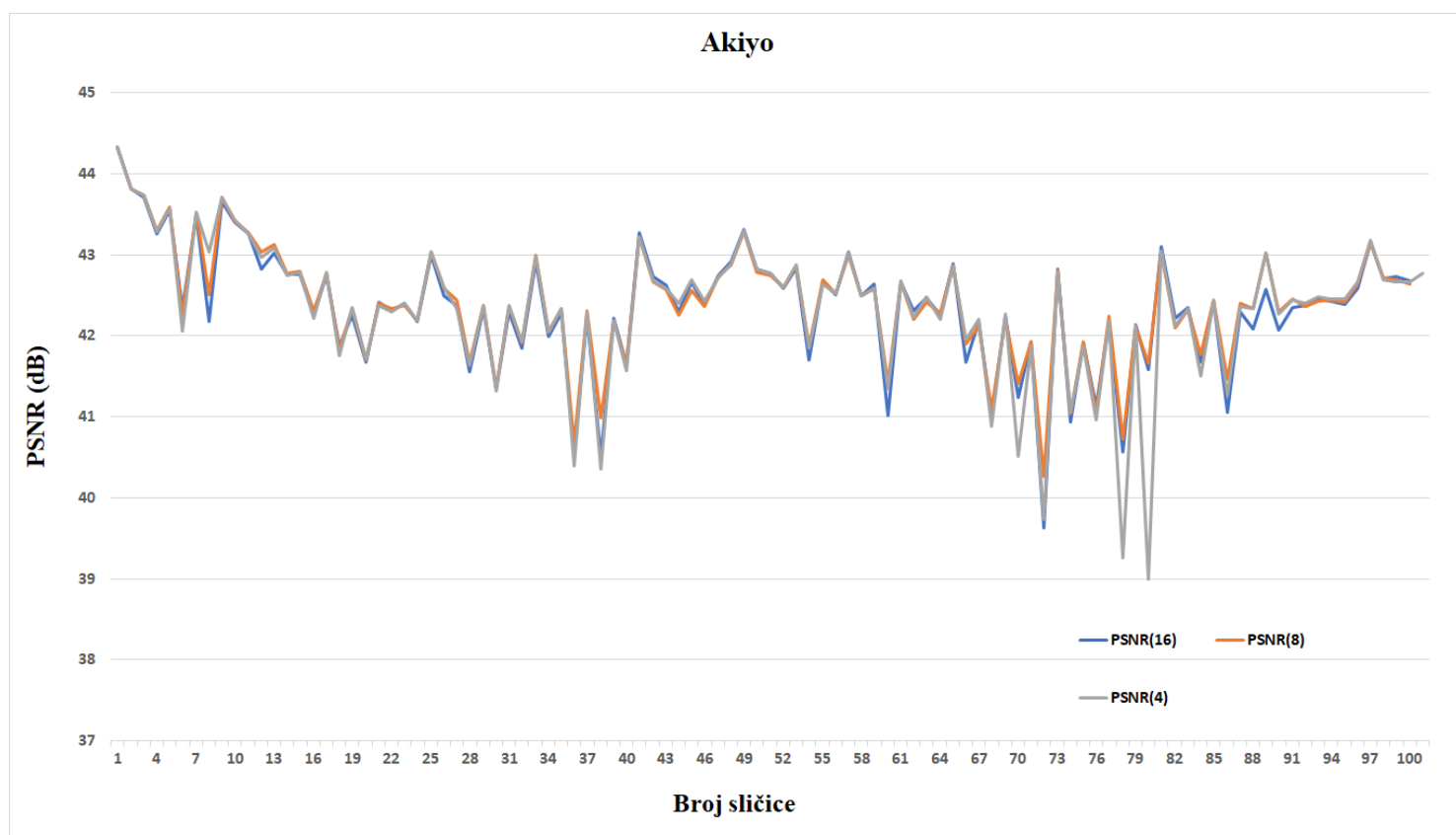


Slika 3.6. Prosječne PSNR vrijednosti sličica videosekvence *Stefan*

Posljednja videosekvenca *Stefan* ima najviše pokreta od svih sekvenci. Kamera i objekt su stalno u pokretu i često prave brze pomake. Zbog toga procjena pokreta ne može uvijek odrediti precizno vektore pomaka. Procjena pokreta ima veliki utjecaj na kvalitetu interpoliranih slika te PSNR vrijednost kod nekih sličica pada i do 22 – 23 dB. PSNR vrijednosti su jako promjenjive, između 22 i 38 dB skoro kroz cijeli graf, što dalje ukazuje na veliku količinu pokreta i da procjena pokreta nije uvijek mogla precizno odrediti vektore pomaka. Za prvih 100 sličica korištenjem veličine bloka 16 x 16 elemenata slike u ES postupku pretrage se najčešće dobivala najniža PSNR vrijednost, dok je korištenje veličine bloka 8 x 8 elemenata slike najčešće rezultiralo najvećom PSNR vrijednosti. To opet pokazuje da je u ES postupku pretrage, veličina bloka 8 x 8 elemenata slike dobar kompromis između veće i manje veličine.

### 3.2. Rezultati za postupak pretrage Diamond Search (DS)

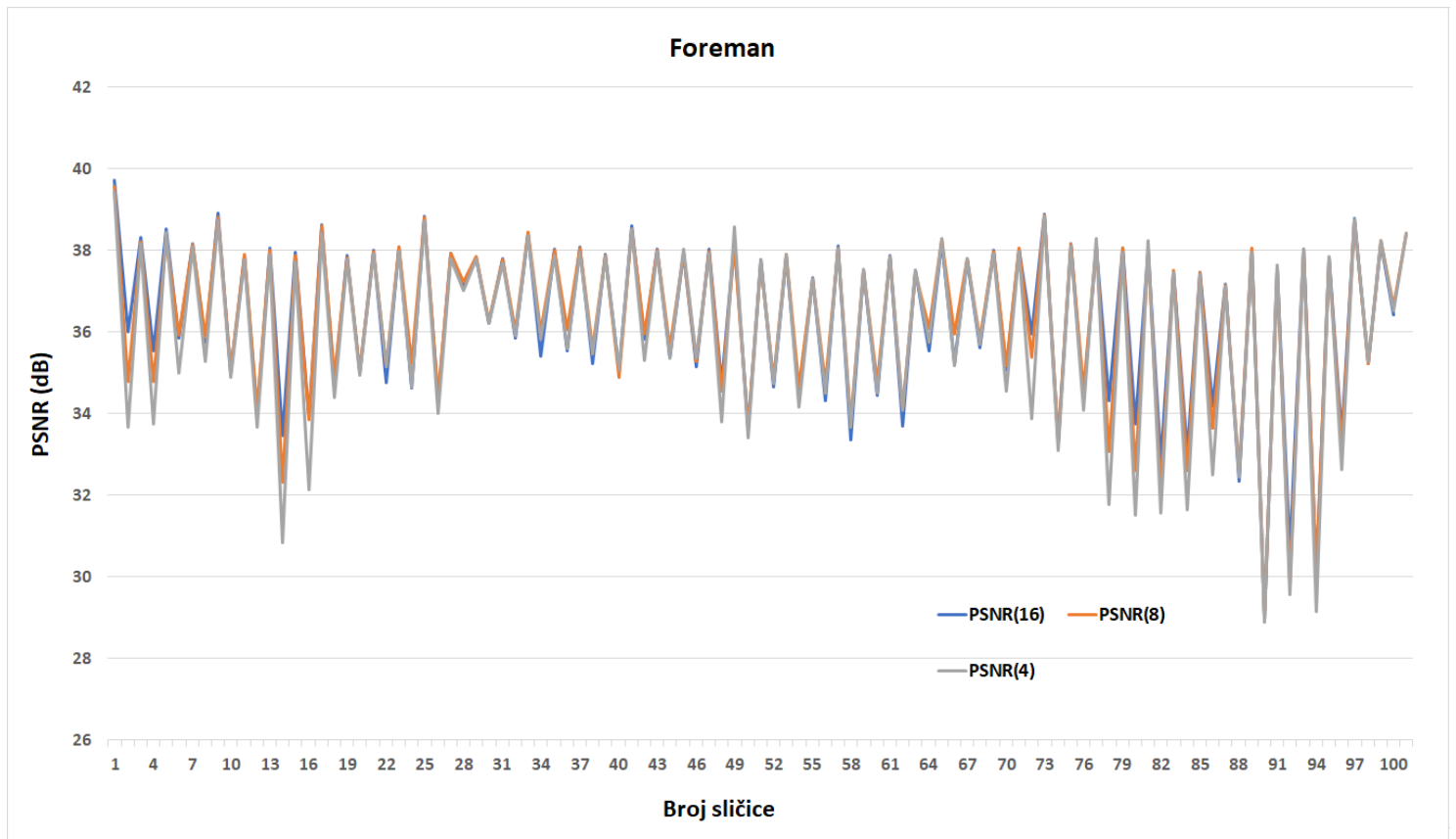
Slijede rezultati PSNR vrijednosti koji su dobiveni uspoređivanjem originalnog videa i videa čija je vremenska rezolucija povećana korištenjem DS postupka pretraživanja za svih pet videosekvenci. Na jednom grafu bit će prikazane prosječne PSNR vrijednosti sličica jedne videosekvence za sve tri testne veličine bloka. Prikazat će se vrijednosti za prvih 100 sličica kako bi graf bio jasniji. Plava linija predstavlja vrijednost PSNR za veličinu bloka 16 x 16 elemenata slike, narančasta za veličinu bloka 8 x 8 elemenata slike, a siva za veličinu bloka 4 x 4 elemenata slike. Prvo je prikazana videosekvencu *Akiyo*:



Slika 3.7. Prosječne PSNR vrijednosti sličica videosekvence *Akiyo*

Kod prve videosekvence kada je korišten postupak pretrage DS, veličina bloka nije imala veliki utjecaj na kvalitetu interpolirane slike. Sekvenca *Akiyo* je dosta mirna, kamera je nepomična i objekt pravi male pokrete te je procjena pokreta mogla precizno odrediti vektore pomaka. PSNR vrijednosti su zbog toga dosta velike za prvih 100 sličica ove sekvence. Najveći pad se dogodio za veličinu bloka 4 x 4 elemenata slike kod 80. sličice što ukazuje da je tu objekt učinio malo veći pomak nego u ostalih 100 sličica te mala veličina bloka u ovom postupku pretrage nije mogla najpreciznije odrediti vektor pomaka.

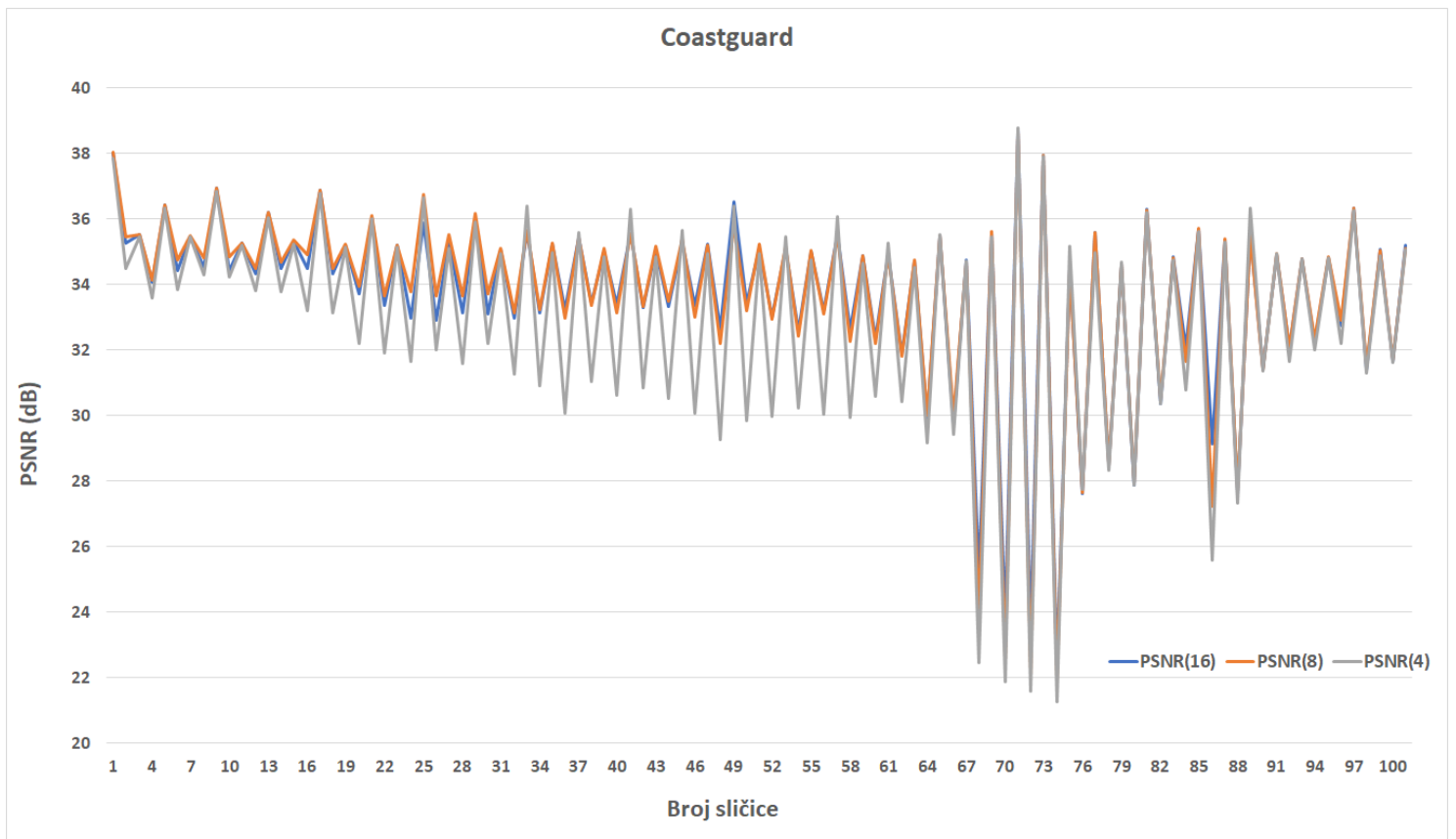
Sljedeće je prikazana videosekvenca *Foreman*:



Slika 3.8. Prosječne PSNR vrijednosti sličica videosekvenca *Foreman*

Videosekvenca *Foreman* ima puno više kretanja jer se i objekt i kamera konstantno pomiču. Zbog toga procjena pokreta ne može toliko precizno odrediti vektore pomaka, a to rezultira PSNR vrijednostima koje su jako promjenjive i nisu toliko visoke kao što su u sekvenci *Akiyo*. Na grafu se za prvih 100 sličica može jasno vidjeti da su se korištenjem veličine bloka 4 x 4 elemenata slike dobile najgore PSNR vrijednosti jer je u skoro svakoj sličici siva linija koja predstavlja tu veličinu najniža. To bi značilo da koristeći najmanju veličinu bloka procjena pokreta u DS postupku pretrage najgore odredi vektore pomaka, što ima smisla s obzirom da sekvenca *Foreman* ima veću količinu kretanja te veličina bloka 4 x 4 elemenata slike propusti neke veće kretanje.

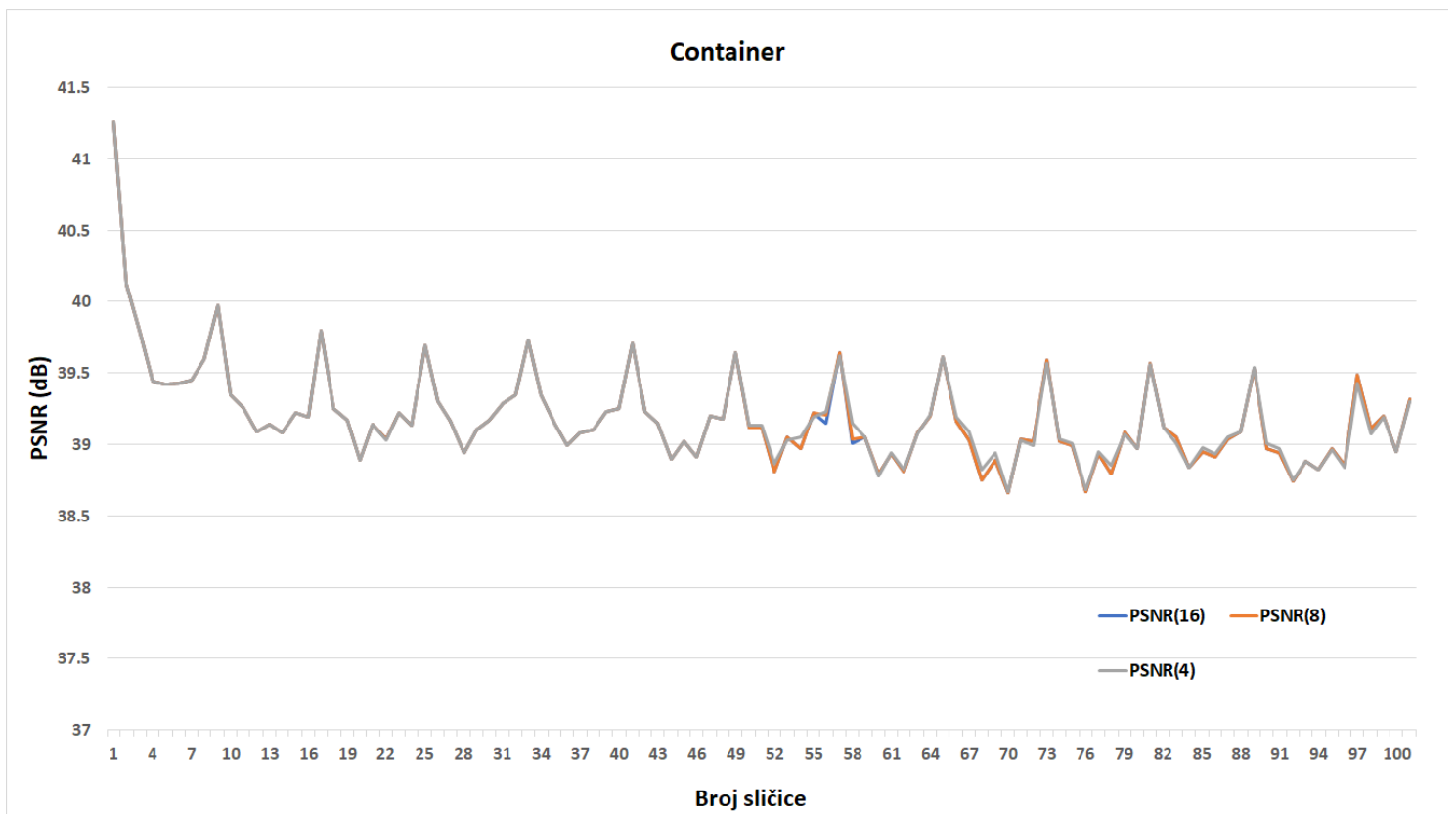
## Videosekvenca *Coastguard*:



*Slika 3.9. Prosječne PSNR vrijednosti sličica videosekvenca *Coastguard**

Videosekvenca *Coastguard* ima pomak dva objekta te u jednom trenutku i veliki pomak kamere. Ovdje se jako dobro vidi utjecaj veličine bloka na kvalitetu interpolirane slike, jer se jasno vidi da su se korištenjem veličine bloka 4 x 4 elemenata slike dobili uvjerljivo najgori rezultati, a korištenjem veličine bloka 8 x 8 elemenata slike najbolji. Najveći padovi i porasti PSNR vrijednosti događaju se na grafu prilikom velikog pomaka kamere, dok su u početku PSNR vrijednosti stalno između 30 i 37 dB. Kamera se naglo i jako pomiče te zbog toga procjena pokreta nije najpreciznija i to ima loš utjecaj na kvalitetu interpoliranih slika kao što se vidi na grafu na sličicama gdje se događaju veliki padovi i do 21 dB. Ovo opet pokazuje da se korištenjem veličine bloka 4 x 4 elemenata slike u DS postupku pretrage jako loše procjene vektora pomaka u sekvencama koje imaju veću količinu kretanja.

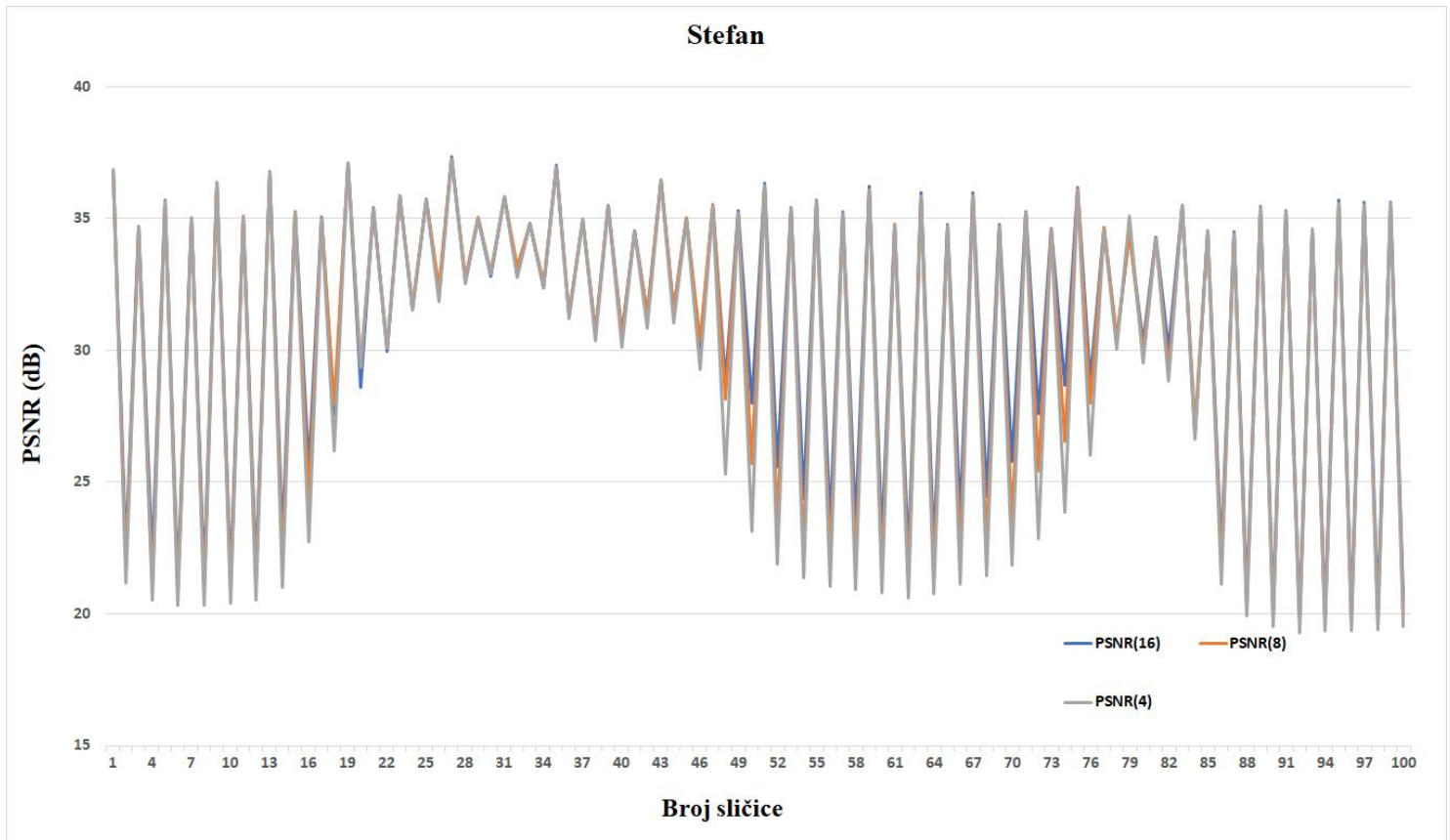
Videosekvenca *Container*:



*Slika 3.10. Prosječne PSNR vrijednosti sličica videosekvence Container*

Ovdje su kao i u ES postupku pretrage vrijednosti prosječnog PSNR-a najslabije dosad. Na grafu se razlike ne vide uvijek, nego samo u nekim sličicama. PSNR vrijednosti su dosta visoke i jako slične jer je videosekvenca s jako malo pokreta te procjena pokreta precizno odredi vektore pomaka koristeći svaku veličinu bloka. Pomoću grafa je teško procijeniti koja veličina bloka ostvaruje najbolje rezultate, no u određenim sličicama se vidi da veličina 4 x 4 elemenata slike ostvari veću kvalitetu nego ostale dvije veličine.

Posljednja videosekvencu *Stefan*:



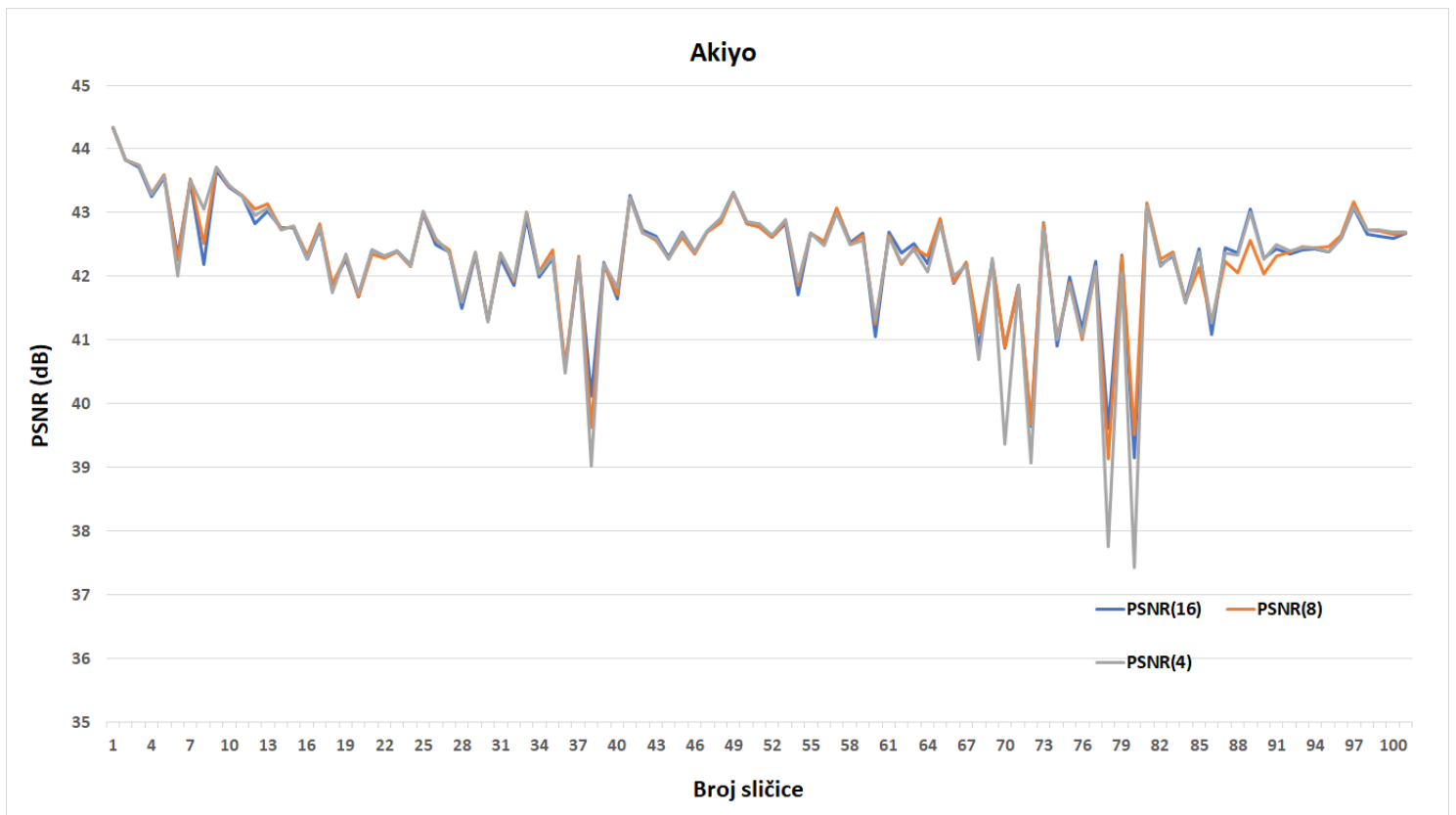
*Slika 3.11. Prosječne PSNR vrijednosti sličica videosekvence Stefan*

Videosekvencu *Stefan* ima najviše pokreta od svih sekvenci. Kamera i objekt su stalno u pokretu i često prave brze pomake. Zbog toga procjena pokreta ne može uvijek odrediti precizne vektore pomaka. Procjena pokreta ima veliki utjecaj na kvalitetu interpoliranih slika te PSNR vrijednost kod nekih sličica pada i ispod 20 dB. PSNR vrijednosti su jako promjenjive, između 20 i 38 dB skoro kroz cijeli graf što dalje ukazuje na veliku količinu pokreta i da procjena pokreta nije uvijek mogla precizno odrediti vektore pomaka. Najveći prosječni PSNR svih sličica u ovoj videosekvenci je bio kada se koristila veličina bloka 16 x 16 elemenata slike, a najmanji kada se koristila veličina bloka 4 x 4 elemenata slike. To se može vidjeti i na grafu gdje je siva linija skoro uvijek najmanje vrijednosti, a plava najveće. To isto ukazuje kao i sekvence *Foreman* i *Coastguard* da u postupku pretrage DS veće veličine bloka kao 16 x 16 elemenata slike ostvaruju puno bolje rezultate nego manje veličine bloka u sekvencama sa velikim pomacima.



### 3.3. Rezultati za postupak pretrage New Three Step Search (NTSS)

Slijede rezultati PSNR vrijednosti koji su dobiveni uspoređivanjem originalnog videa i videa čija je vremenska rezolucija povećana korištenjem NTSS postupka pretraživanja za svih pet videosekvenci. Na jednom grafu bit će prikazane prosječne PSNR vrijednosti sličica jedne videosekvence za sve tri testne veličine bloka. Prikazat će se vrijednosti za prvih 100 sličica kako bi graf bio jasniji. Plava linija predstavlja vrijednost PSNR za veličinu bloka 16 x 16 elemenata slike, narančasta za veličinu bloka 8 x 8 elemenata slike, a siva za veličinu bloka 4 x 4 elemenata slike. Prvo je prikazana videosequencija *Akiyo*:

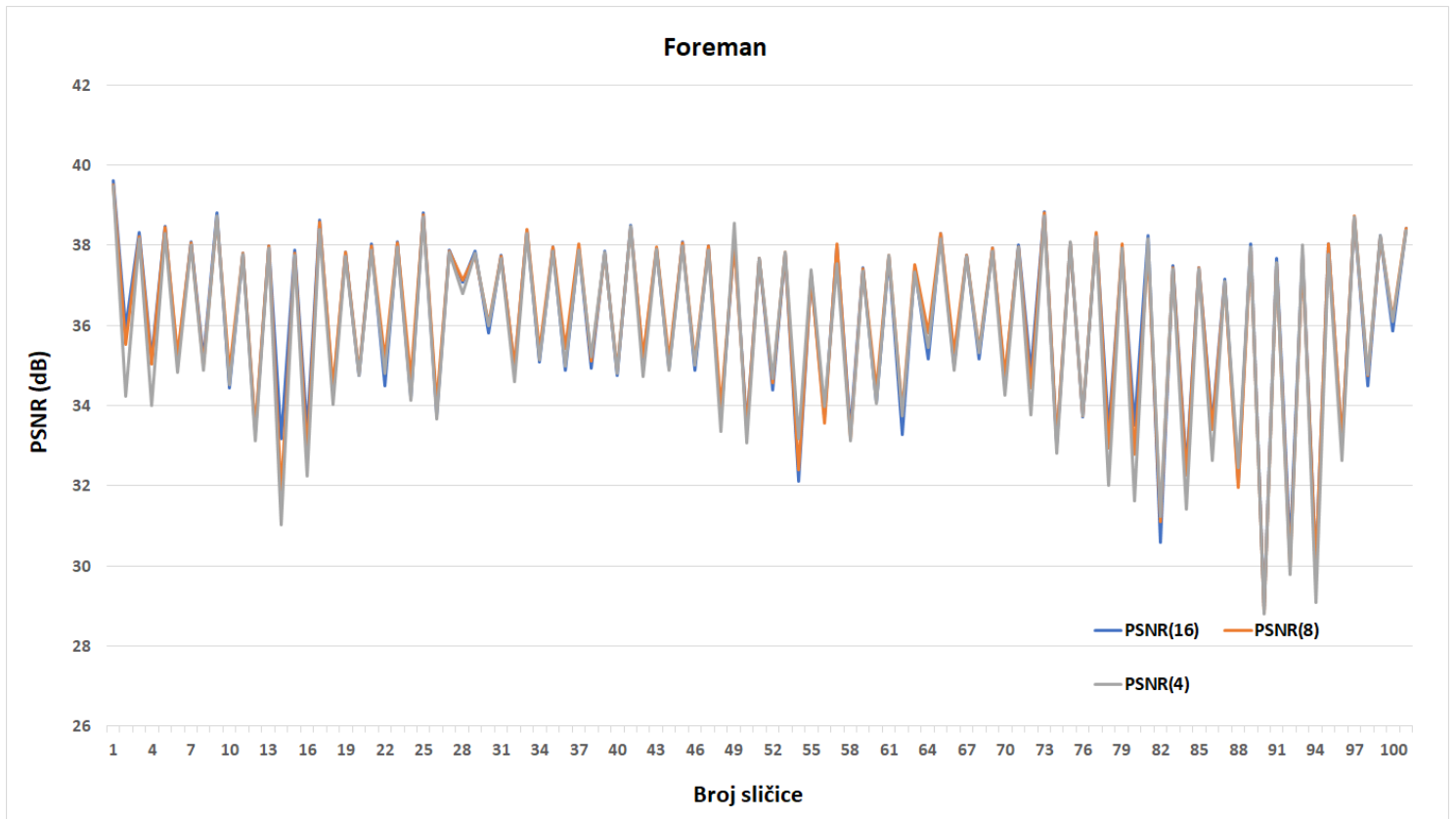


Slika 3.12. Prosječne PSNR vrijednosti sličica videosekvence *Akiyo*

Prosječne PSNR vrijednosti interpoliranih sličica jako su slične za sve tri veličine bloka. Graf se ponaša slično kao i kod prva dva postupka pretrage, no ovdje imamo najviše padova PSNR vrijednosti ispod 40 dB, a kod tih padova se najjasnije vidi da se korištenjem veličine bloka 4 x 4 elemenata slike dobije najmanja kvaliteta interpolirane slike. PSNR vrijednosti su kroz cijeli graf dosta visoke, između 37 i 44 dB, jer je videosequencija jako mirna, bez pomaka kamere i sa jako malo pomaka objekta. Zbog toga možemo zaključiti da je procjena pokreta, koja ima veliki utjecaj

na PSNR vrijednosti, odrađena dosta precizno što je i za očekivati kod mirnog sadržaja kakav je u ovoj sekvenci.

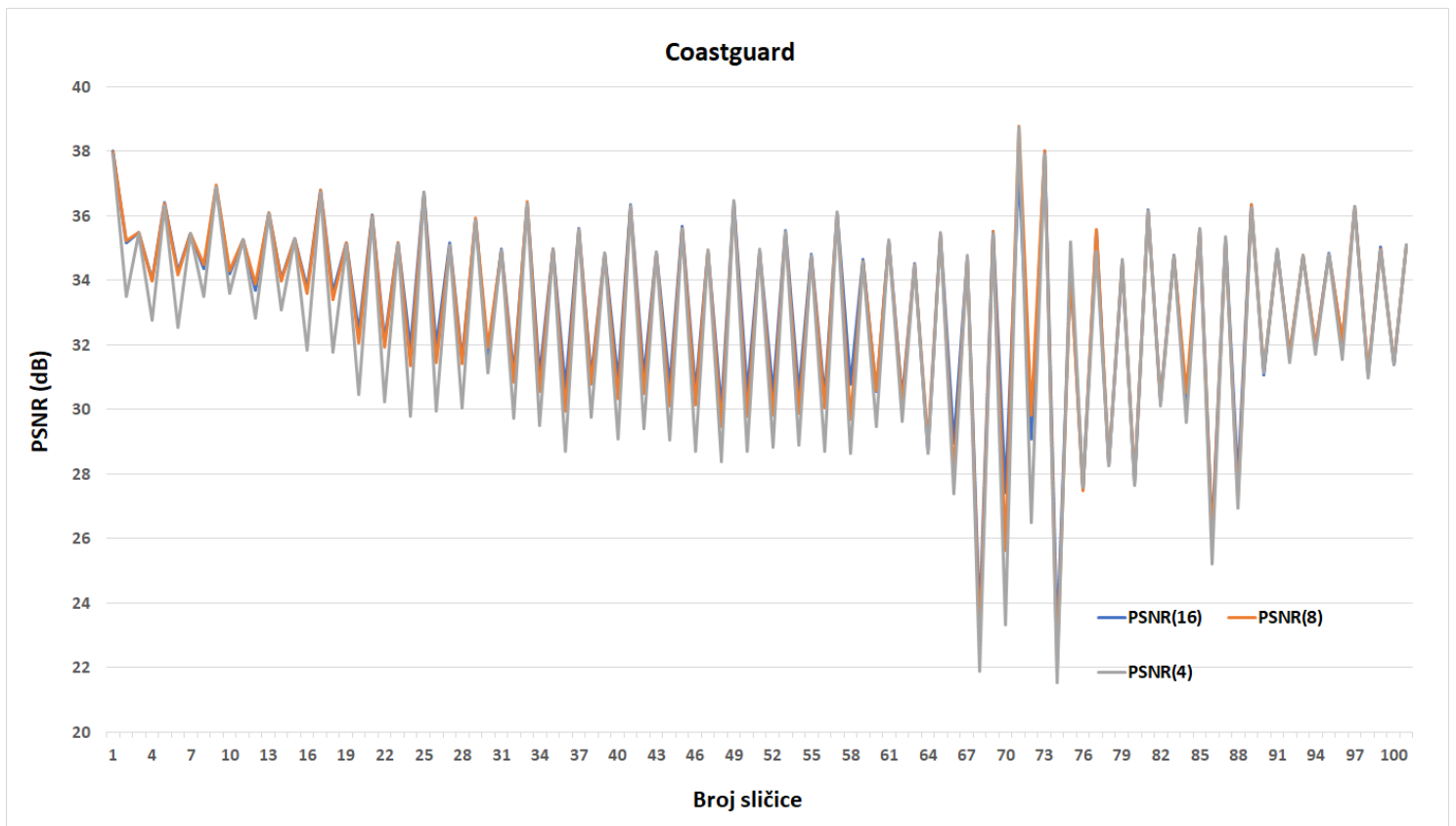
Sekvenca *Foreman*:



Slika 3.13. Prosječne PSNR vrijednosti sličica videosekvence *Foreman*

U videosekvenci *Foreman* imamo više pomaka ali su isto male razlike u PSNR vrijednostima. PSNR vrijednosti su kao i kod prva dva postupka pretrage jako promjenjive zbog puno pokreta kamere i objekta. Zbog toga procjena pokreta nije odrađena toliko precizno kao u mirnijoj *Akiyo* sekvenci, te su PSNR vrijednosti manje i s većim padovima u vrijednosti.

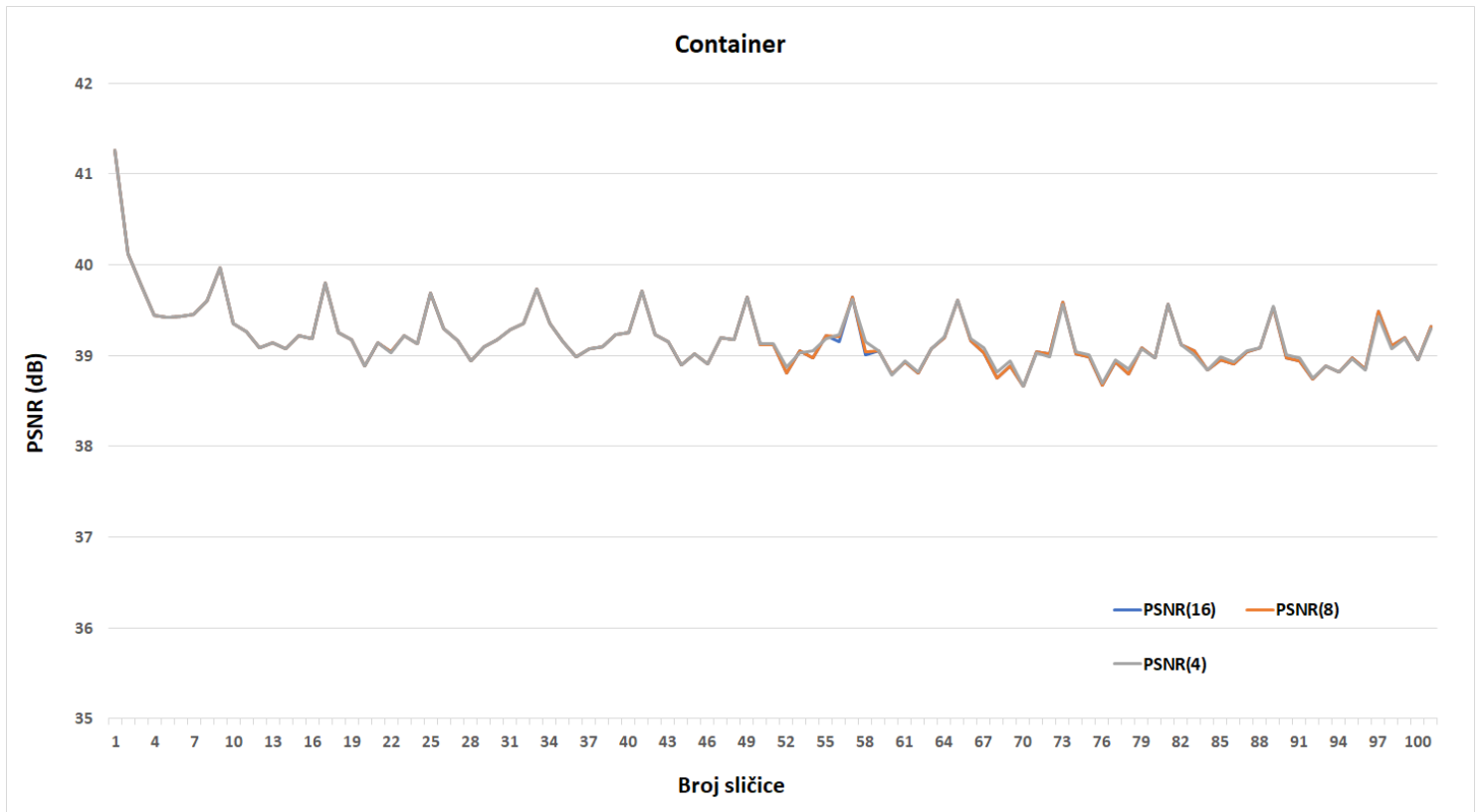
## Sekvenca *Coastguard*:



Slika 3.14. Prosječne PSNR vrijednosti sličica videosekvence *Coastguard*

U ovoj videosekvenci se malo bolje vide razlike u kvaliteti za svaku veličinu bloka. PSNR vrijednosti su jako promjenjive kroz cijelu sekvenču i nisu toliko velike kao u mirnoj *Akiyo* sekvenču, a najviše padaju kada se pomiče kamera što se vidi i na grafu oko 70. sličice. PSNR vrijednost tada najviše pada jer zbog brzog pomaka kamere procjena pokreta ne može najpreciznije odrediti vektore pomaka a to ima veliki utjecaj na kvalitetu interpolirane slike. Jasno se vidi da se korištenjem veličine bloka 4 x 4 elemenata slike skoro kroz cijeli graf dobije najmanja PSNR vrijednost. Za prvih 100 sličica na grafu, veličina bloka 16 x 16 elemenata slike je najčešće imala najveću PSNR vrijednost. To ukazuje da i u NTSS postupku pretrage kao i u DS postupku korištenje male veličine bloka kao 4 x 4 elemenata slike rezultira lošijom kvalitetom interpolirane slike u bržim sekvenčama.

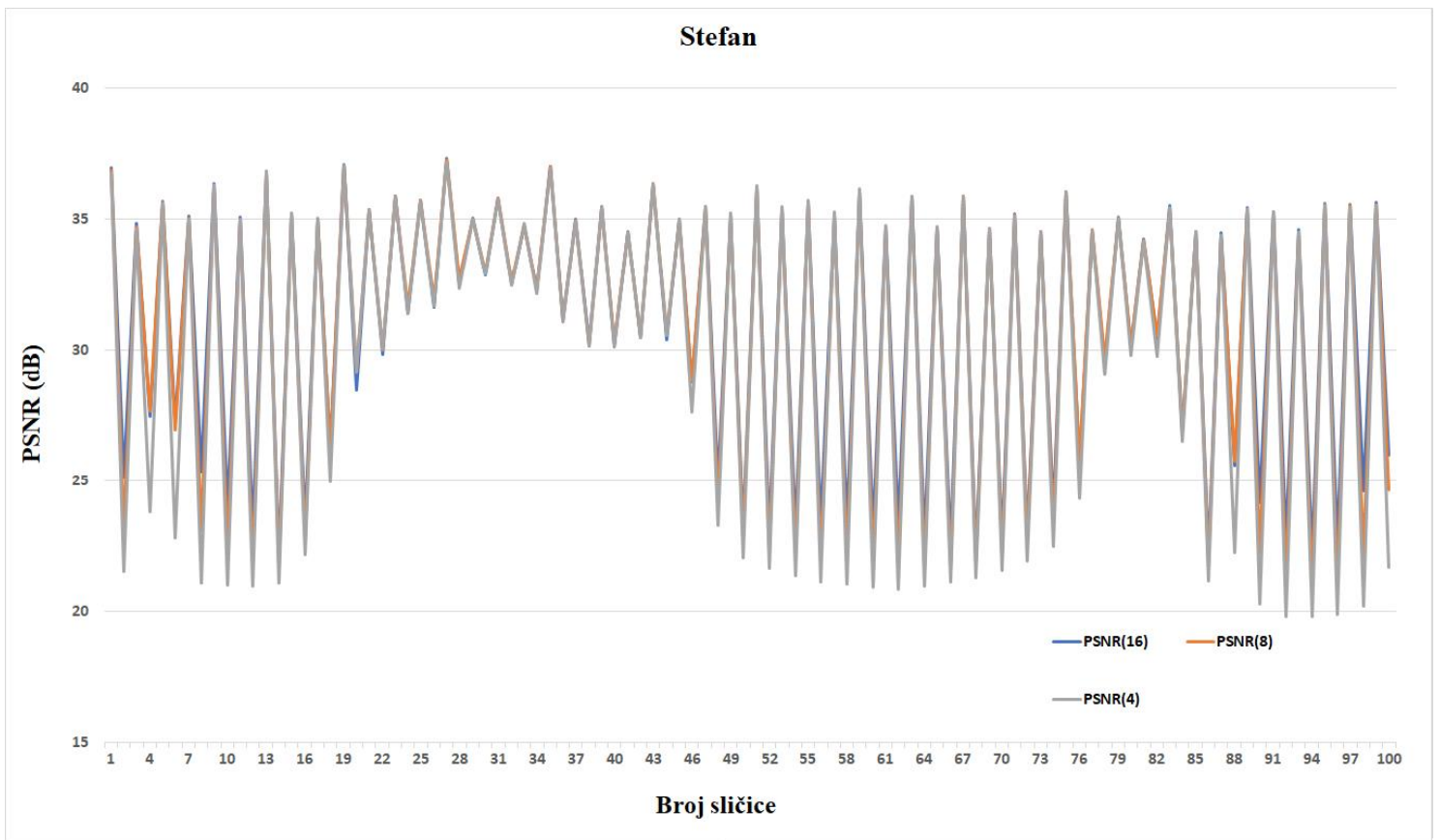
## Sekvenca *Container*:



*Slika 3.15. Prosječne PSNR vrijednosti sličica videosekvenca Container*

Ovdje su prosječne PSNR vrijednosti sličica jako slične i teško je vidjeti razliku na grafu kao i u prva dva postupka pretrage. PSNR vrijednosti su kroz cijeli graf jako slične i visoke jer videosekvenca nema puno pomaka te je procjena pokreta jako precizno odredila vektore pomaka za svaku veličinu bloka. Najveća kvaliteta interpoliranih sličica je ostvarena kada se koristila veličina bloka 4 x 4 elemenata slike.

Posljednja sekvenca *Stefan*:



*Slika 3.16. Prosječne PSNR vrijednosti sličica videosekvence Stefan*

I u NTSS-u je slična situacija s videosekvencom *Stefan* koja ima najviše pokreta od svih sekvenci. Kamera i objekt su stalno u pokretu i često prave brze pomake. Zbog toga procjena pokreta ne može uvijek odrediti precizne vektore pomaka. Procjena pokreta ima veliki utjecaj na kvalitetu interpoliranih slika te PSNR vrijednost kod nekih sličica pada i ispod 20 dB. Cijeli graf je jako promjenjiv i često pada i raste između 20-ak dB i oko 38 dB što dalje upućuje na veliku količinu pokreta u sekvenci. Na grafu se vidi da se najmanji PSNR za prvih 100 sličica dobije kada se koristi veličina bloka 4 x 4 elemenata slike, što je i za očekivati kod brze sekvence kao *Stefan* nakon što smo dobili slične rezultate i za ostale dvije brze sekvence *Foreman* i *Coastguard*.

### 3.4. Usporedba rezultata

Za videosekvence koje su testirane može se zaključiti da se korištenjem veličine bloka 8 x 8 elemenata slike najčešće postižu najbolji rezultati. Najveća testirana veličina bloka, 16 x 16 elemenata slike, zbog svoje veličine najbrže provjeri sve blokove i može precizno procijeniti neke veće pomake, no zbog toga propusti manje kretnje. Najmanja veličina bloka, 4 x 4 elemenata slike, treba najviše vremena da se izvrši procjena, može dobro procijeniti male kretnje, ali propusti velike pomake objekata. Sve to znači da je veličina bloka 8 x 8 elemenata slike, za ovu bazu sekvenci koje su testirane dobar kompromis za male kretnje koje propusti veličina bloka 16 x 16 elemenata slike i velike kretnje koje propusti veličina bloka 4 x 4 elemenata slike. Ovi rezultati također ukazuju i na to da u bržim sekvencama kao što su *Foreman*, *Coastguard* i *Stefan* bolji su rezultati češće dobiveni kada se koristila veća veličina bloka 8 x 8 elemenata slike ili 16 x 16 elemenata slike, pogotovo kod DS i NTSS postupka pretrage. S druge strane, kod mirnijih sekvenci kao *Akiyo* i *Container* malo bolje rezultate imale su manje veličine bloka kao 4 x 4 elemenata slike i 8 x 8 elemenata slike. Na sljedećim tablicama 3.1., 3.2. i 3.3. prikazane su prosječne vrijednosti svih sličica za sve postupke pretrage koji su bili implementirani.

*Tablica 3.1. Prosječne PSNR vrijednosti svih sličica za postupak pretrage ES*

Veličina bloka	Prosječni PSNR svih sličica [dB]				
	ES				
	Akiyo	Foreman	Coastguard	Container	Stefan
<b>16 x 16</b>	42.35	35.06	34.15	38.67	30.73
<b>8 x 8</b>	42.38	35.45	34.44	38.68	31.28
<b>4 x 4</b>	42.38	35.03	34.35	38.68	30.87

*Tablica 3.2. Prosječne PSNR vrijednosti svih sličica za postupak pretrage DS*

Veličina bloka	Prosječni PSNR svih sličica [dB]				
	DS				
	Akiyo	Foreman	Coastguard	Container	Stefan
<b>16 x 16</b>	42.34	34.99	34.19	38.67	29.78
<b>8 x 8</b>	42.38	34.83	34.24	38.67	29.53
<b>4 x 4</b>	42.35	34.57	33.5	38.68	29.16

*Tablica 3.3. Prosječne PSNR vrijednosti svih sličica za postupak pretrage NTSS*

Veličina bloka	Prosječni PSNR svih sličica [dB]				
	NTSS				
	Akiyo	Foreman	Coastguard	Container	Stefan
<b>16 x 16</b>	42.33	34.93	33.46	38.67	29.74
<b>8 x 8</b>	42.35	34.79	33.25	38.67	29.48
<b>4 x 4</b>	42.31	34.52	32.81	38.68	29.11

Rezultati PSNR vrijednosti po sekvencama su sljedeći. Videosekvencu *Akiyo* je u DS i NTSS postupku pretrage imala najveći prosječni PSNR svih sličica kada je veličina bloka bila postavljena na 8 x 8 elemenata slike. U ES postupku pretrage prosječni PSNR svih sličica bio je jednak kod veličine bloka 4 x 4 elemenata slike i 8 x 8 elemenata slike. Na slici 3.17. bit će prikazana određena originalna i interpolirana sličica kada se koristio DS postupak pretrage za sve testirane veličine blokova.



(a)



(b)



(c)



(d)

**Slika 3.17.** Prikaz 8. originalne i interpolirane sličice videosekvence *Akiyo* (a) Originalna (b) Veličina bloka = 16 (PSNR=42.18 dB) (c) Veličina bloka = 8 (PSNR=42.51 dB) (d) Veličina bloka = 4 (PSNR=43.03 dB)



Videosekvenca *Foreman* je u DS i NTSS postupku pretrage imala najveći prosječni PSNR svih sličica kada je veličina bloka bila postavljena na 16 x 16 elemenata slike, a u ES postupku pretrage kada je veličina bloka bila postavljena na 8 x 8 elemenata slike. Na slici 3.18. bit će prikazana određena originalna i interpolirana sličica kada se koristio NTSS postupak pretrage za sve testirane veličine blokova.



(a)



(b)



(c)



(d)

**Slika 3.18.** Prikaz 16. originalne i interpolirane sličice videosekvence *Foreman* (a) Originalna (b) Veličina bloka = 16 (PSNR=33.37 dB) (c) Veličina bloka = 8 (PSNR=32.87 dB) (d) Veličina bloka = 4 (PSNR=32.25 dB)

Videosekvencu *Coastguard* je u ES i DS postupku pretrage imala najveći prosječni PSNR svih sličica kada je veličina bloka bila na 8 x 8 elemenata slike, a u NTSS postupku pretrage kada je veličina bloka bila postavljena na 16 x 16 elemenata slike. Na slici 3.19. bit će prikazana određena originalna i interpolirana sličica kada se koristio ES postupak pretrage za sve testirane veličine blokova.



(a)



(b)



(c)



(d)

**Slika 3.19.** Prikaz 33. originalne i interpolirane sličice video sekvence *Coastguard*  
(a) Originalna (b) Veličina bloka = 16 (PSNR=35.69 dB) (c) Veličina bloka = 8  
(PSNR=35.7 dB) (d) Veličina bloka = 4 (PSNR=36.79 dB)

Videosekvenca *Container* je u DS i NTSS postupku pretrage imala najveći prosječni PSNR svih sličica kada je veličina bloka bila postavljena na 4 x 4 elemenata slike, dok je u ES postupku pretrage PSNR bio jednak kod veličine bloka 8 x 8 elemenata slike i 4 x 4 elemenata slike. Na slici 3.20. bit će prikazana određena originalna i interpolirana sličica kada se koristio DS postupak pretrage za sve testirane veličine blokova.



(a)



(b)



(c)



(d)

**Slika 3.20.** Prikaz 58. originalne i interpolirane sličice videosekvence *Container* (a) Originalna (b) Veličina bloka = 16 (PSNR=39.01 dB) (c) Veličina bloka = 8 (PSNR=39.04 dB) (d) Veličina bloka = 4 (PSNR=39.15 dB)



Videosekvencu *Stefan* je u DS i NTSS postupku pretrage imala najveći prosječni PSNR svih sličica kada je veličina bloka bila postavljena na 16 x 16 elemenata slike. U ES postupku pretrage najveći prosječni PSNR svih sličica bio je kod veličine bloka 8 x 8 elemenata slike. Na slici 3.21. bit će prikazana određena originalna i interpolirana sličica kada se koristio NTSS postupak pretrage za sve testirane veličine blokova.



(a)



(b)



(c)



(d)

**Slika 3.21.** Prikaz 92. originalne i interpolirane sličice videosekvence *Stefan* (a) Originalna (b) Veličina bloka = 16 (PSNR=22.69 dB) (c) Veličina bloka = 8 (PSNR=21.24 dB) (d) Veličina bloka = 4 (PSNR=19.81 dB)

## 4. ZAKLJUČAK

Radom je analiziran utjecaj veličine bloka prilikom procjene pokreta na kvalitetu interpolirane slike koja se dodaje u videozapis kod povećanja vremenske rezolucije. Povećanje vremenske rezolucije je važan postupak jer danas često dolazi do smanjivanja videozapisa uklanjanjem određenih sličica kako bi se video mogao lakše poslati preko interneta. Postoji mnogo metoda povećanja vremenske rezolucije, a i mnogo različitih algoritama za procjenu pokreta koja je bitan dio povećanja vremenske rezolucije. Svaki od njih ima prednosti i mane, i neki su opisani u radu.

Zadatak je izvršen pomoću programskog paketa FFmpeg. U njemu je smanjena vremenska rezolucija korištenih testnih videosekvenci, potom povećana MCFI algoritmom s postavljenim parametrima (vrsta procjene pokreta, postupak pretrage područja pretrage, veličina bloka) i zatim su izračunate prosječne PSNR vrijednosti sličica usporedbom sličica originalnog videosignala i videosignala kojem je vremenska rezolucija povećana. Za povećanje vremenske rezolucije koristila se dvosmjerna procjena pokreta te tri različita načina pretraživanja područja pretrage i tri veličine bloka. Postupci su pretrage bili: iscrpna pretraga (eng. *Exhaustive Search – ES*), dijamantna pretraga (eng. *Diamond Search – DS*), nova pretraga u tri koraka (eng. *New Three Step Search – NTSS*). Veličine bloka čiji se utjecaj na kvalitetu interpolirane slike testirao su bile: 16 x 16 elemenata slike, 8 x 8 elemenata slike, 4 x 4 elemenata slike. Kvaliteta interpolirane slike bila je procijenjena navedenom PSNR metodom.

Analizom rezultata može se zaključiti da za ovih 5 testiranih sekvenci veličina bloka nema veliki utjecaj na kvalitetu interpolirane slike. Primjerice, najveća razlika u prosječnoj PSNR vrijednosti svih sličica bila je u videosekvenci *Coastguard* kada se koristio DS postupak pretrage, gdje se korištenjem veličine bloka 8 x 8 elemenata slike dobio prosječni PSNR svih sličica za 0.69 dB veći nego kada se koristila veličina bloka 4 x 4 elemenata slike. Analiza prosječnih PSNR vrijednosti svih sličica je ukazala na to da veličina bloka 8 x 8 elemenata slike najčešće daje najbolje rezultate te da je ova veličina dobar kompromis između najveće i najmanje veličine bloka za testirane sekvence jer veća veličina bloka može ponekad propustiti ili lošije procijeniti manji pomak dok za manju veličinu bloka vrijedi obratno. Isto tako, može se zaključiti da su videosekvence kao *Akiyo* i *Container*, gdje je kamera bila potpuna mirna i nije bilo puno velikih pokreta, imale nešto bolje PSNR rezultate kada se koristila manja veličina bloka 4 x 4 elemenata slike ili 8 x 8 elemenata slike. S druge strane, kod videosekvenci *Foreman*, *Coastguard* i *Stefan*, gdje je postojalo pomicanje kamere i više pokreta objekata, postizale su se bolje prosječne PSNR vrijednosti kada se koristila veća veličina bloka 8 x 8 elemenata slike ili 16 x 16 elemenata slike,

ovo se pogotovo vidi kod DS i NTSS postupka pretrage gdje su se korištenjem veličine bloka 4 x 4 elemenata slike uvijek dobivali najgori PSNR rezultati kod ovih sekvenci dok je u ES postupku bilo iznimki.

## POPIS LITERATURE

- [1] S. Rimac-Drlje, D. Vranješ, Fast frame-rate up-conversion method for video enhancement: 2016 International Conference on Systems, Signals and Image Processing (IWSSIP), 2016, str. 1-4, dostupno na: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7502736> [7.9.2021]
- [2] D. Vranješ, S. Rimac-Drlje, M. Vranješ, Influence of block size on motion vector estimation error in enhancement of video temporal resolution: 2017 International Conference on Smart Systems and Technologies (SST), 2017, str. 263-267, dostupno na: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8188707> [7.9.2021]
- [3] A. Barjatya, Block Matching Algorithms for Motion Estimation: 2004, dostupno na: <https://www.mathworks.com/matlabcentral/fileexchange/8761-block-matching-algorithms-for-motion-estimation> [7.9.2021]
- [4] D. D. Bhavsar, R. N. Gonawala, Three Step Search Method for Block Matching Algorithm: dostupno na: [https://www.digitalxplore.org/up\\_proc/pdf/62-1397565973101-104.pdf](https://www.digitalxplore.org/up_proc/pdf/62-1397565973101-104.pdf) [7.9.2021]
- [5] R. Li, B. Zeng, M. L. Liou, A New Three-Step Search Algorithm for Block Motion Estimation: IEEE Transactions on Circuits and Systems for Video Technology, vol. 4, no. 4, str. 438-442, Kolovoz 1994, dostupno na: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=313138> [7.9.2021]
- [6] S. Zhu, K. Ma, A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation: IEEE Transactions on Image Processing, vol. 9, no. 2, str. 287-290, Veljača 2000, dostupno na: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=821744> [7.9.2021]
- [7] Članak: Peak Signal-to-Noise Ratio as an Image Quality Metric, dostupno na: <https://www.ni.com/en-rs/innovations/white-papers/11/peak-signal-to-noise-ratio-as-an-image-quality-metric.html> [7.9.2021]
- [8] Commonly used video test sequences in 4:2:0 YUV format: dostupno na: <http://trace.eas.asu.edu/yuv/> [7.9.2021]
- [9] Video test sequences in uncompressed YUV4MPEG format: dostupno na: <https://media.xiph.org/video/derf/> [7.9.2021]

## SAŽETAK

Razvojem tehnologija porasla je kvaliteta videozapisa a time i njihova veličina. Zbog toga se javlja želja za komprimiranjem videozapisa radi bržeg prijenosa putem interneta ili lakšeg spremanja na medije za pohranu. Ponekad se i smanjuje vremenska i/ili prostorna rezolucija videa u svrhu smanjivanja veličine koja se onda na prijemnoj strani mora povećati kako bi kvaliteta videa bila što bolja, iako nikada ne može dostići originalnu. Za povećanje vremenske rezolucije koriste se MCFI algoritmi čiji je bitan dio procjena pokreta, proces određivanja vektora pokreta između susjednih sličica u videozapisu. Za svrhu procjene pokreta često se koriste algoritmi za pronalazak odgovarajućeg bloka (eng. *Block-matching algorithm – BMA*) koji su opisani u radu. Zadatak je bio odrediti kakav utjecaj ima veličina bloka na kvalitetu interpolirane slike u postupku povećanja vremenske rezolucija. Utjecaj bloka testiran je na tri različita postupka pretrage i pet različitih sekvenci. Zadatak je odrađen pomoću FFmpeg programskog paketa. Taj je utjecaj prikazan pomoću PSNR metode kojom se može mjeriti kvaliteta interpoliranih slika. Kao rezultat dobiveno je da veličina bloka nema veliki utjecaj na kvalitetu interpolirane slike na korištenoj testnoj bazi videosekvenci, no pokazalo se da veća veličina bloka malo bolje djeluje kod videozapisa sa većom količinom pokreta objekata i kada se kamera pomiče, dok manja veličina bloka daje nešto bolje rezultate kod mirnijih videozapisa.

Ključne riječi: blok, interpolacija, procjena pokreta, slika, videozapis



# **INFLUENCE OF BLOCK SIZE FOR MOTION ESTIMATION ON INTERPOLATED IMAGE QUALITY IN THE PROCESS OF INCREASING VIDEO TIME RESOLUTION**

## **ABSTRACT**

With the development of technology, video quality is improving and because of that videos are getting bigger in size. So, to be able to transfer videos over the internet or save storage space when saving them, there is a need for video compression. Sometimes the temporal resolution of the video is lowered to reduce the size even further. The temporal resolution then also needs to be increased on the receiving side to increase the quality of the video, although it can never reach the quality of the original. One of the main parts of frame interpolation algorithms which are used for increasing the temporal resolution of the video is motion estimation, the process of determining motion vectors between neighboring frames in the video. For the purposes of motion estimation, we use block-matching algorithms which are described in the paper. The assignment for this paper was to determine what kind of effect the block size has in the process of motion estimation on the quality of the interpolated picture when increasing the temporal resolution of the video. This effect was tested on three different block search methods and five different video sequences. The assignment was done by using FFmpeg and the quality of the interpolated picture was measured by PSNR. The result showed that block size doesn't have a large effect on the quality of interpolated picture on the test sequences used in the paper, but it did show that bigger block sizes give somewhat better results in video sequences which have larger motions and camera movement, while lower block sizes work better for video sequences with a low amount of motion.

Key words: block, interpolation, motion estimation, frame, video