Lab2 izvještaj

Uvod

U ovoj laboratorijskoj vježbi analizirana je izvedba programa za obradu video podataka paraleliziranog korištenjem OpenMP tehnologije. Program je pokretan na superračunalu BURA, kao i na lokalnom laptop računalu, s ciljem procjene ubrzanja i efikasnosti izvođenja pri različitom broju procesorskih jezgri.

Eksperiment

Za pregled i vizualnu provjeru video podataka u YUV formatu koristio sam program YUView. Ovaj alat omogućava pregled pojedinačnih frameova iz .yuv datoteka te jednostavno dohvaćanje screenshota za potrebe analize kvalitete slike i dokumentacije.

Slika 1: raw_video.yuv

Slika 2: rgb2yuv.yuv

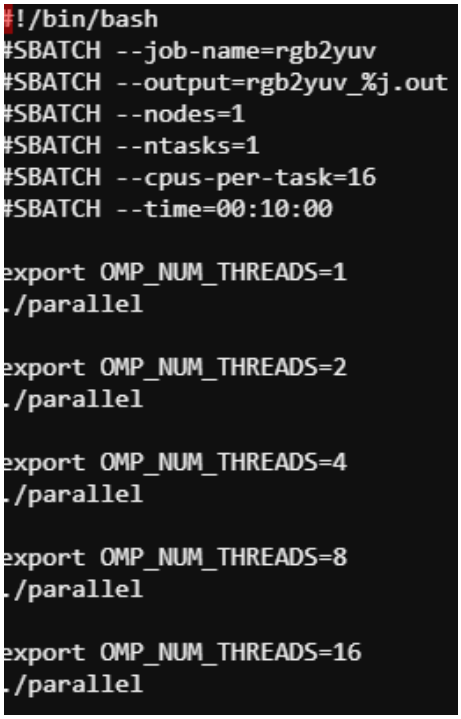
Slika 3: subsampled.yuv

Slika 4: upsampled.yuv

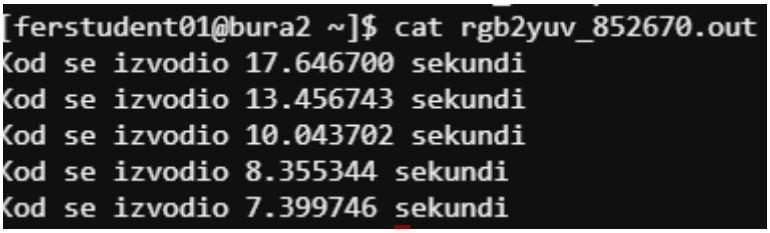
Za testiranje paralelizacije korišten je program parallel.c, koji je preveden s GCC kompajlerom uz opciju -fopenmp. Za usporedbu korištena je i sekvencijska verzija programa sequence.c. Ulazni video podaci su u formatu YUV s rezolucijom 4K.

Pokretanje na superračunalu BURA izvršeno je putem Slurm batch skripte rgb2yuv.sh, koju sam napisao za jednostavno upravljanje zadacima. Skriptu sam pokrenuo naredbom sbatch rgb2yuv.sh. Izlazni podaci vremena izvođenja pohranjeni su u datoteku rgb2yuv_<job_id>.out.

Sekvencijska verzija koda daje približne rezultate kao paralelna koja koristi broj procesora 1, pa je ta paralelna korištena kao referentna (baseline) za računanje ubrzanja.



Slika 5: rgb2yuv.sh



Slika 6: Ispis rgb2yuv_<job_id>.out

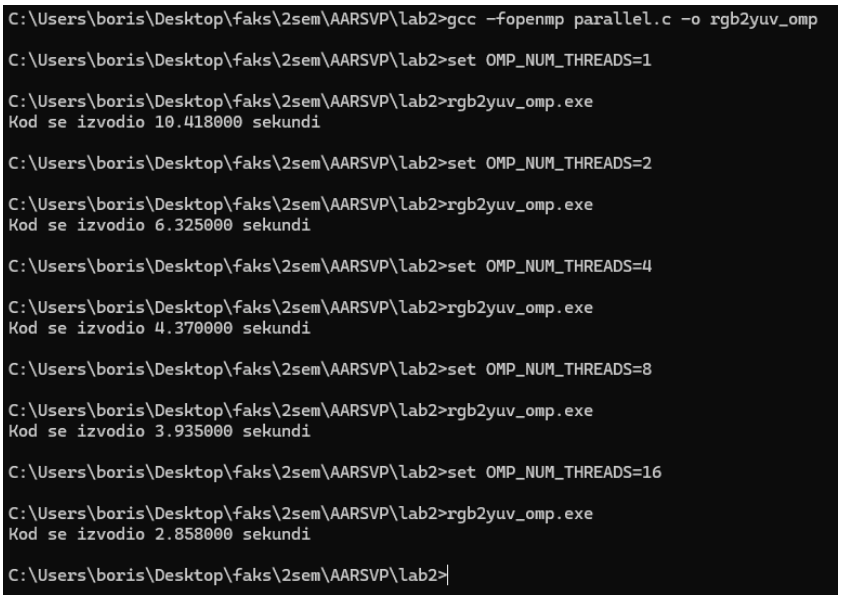
Rezultati i analiza

Superračunalo BURA:

Broj procesora	Vrijeme izvođenja	Ubrzanje	Efikasnost
1	17.646700 s	1x	100%
2	13.456743 s	1.311x	65.55%
4	10.043702 s	1.757x	43.93%
8	8.355344 s	2.112x	26.4%
16	7.399746 s	2.385x	14.91%

Iz dobivenih rezultata vidljivo je da se ubrzanje programa povećava s brojem procesora, no efikasnost paralelizacije brzo opada. Najveći dio dobitka se ostvaruje do 4 jezgre, dok daljnje povećavanje broja jezgri donosi sve manje poboljšanja.

Laptop računalo:



Slika 7: pokretanje na laptopu

Broj procesora	Vrijeme izvođenja	Ubrzanje	Efikasnost
1	10.418000 s	1x	100%
2	6.325000 s	1.647x	82.35%
4	4.370000 s	2.384x	59.6%
8	3.935000 s	2.648x	33.1%
16	2.858000 s	3.64x	22.75%

Na laptopu ubrzanje je bolje i efikasnost se sporije smanjuje nego na BURA-i, što može biti posljedica različite arhitekture, cachea i drugih faktora.

Zaključak

Paralelizacija koda s OpenMP-om donosi značajno ubrzanje, ali se efikasnost smanjuje sa sve većim brojem jezgri, osobito na superračunalu BURA. Iako se ukupno vrijeme smanjuje, gubici efikasnosti sugeriraju moguća uska grla, poput sinkronizacija, dijeljenja memorije ili drugih resursa. Daljnja optimizacija i profiliranje koda mogu pomoći u boljem iskorištavanju dostupnih procesorskih jezgri.

Student: Boris Boronjek

JMBAG: 0036531473