# untitled

Naziv predmeta:  
**Osnovi Algoritama i Struktura DSP 2**

**Projektni zadatak 2**

|  |  |
| --- | --- |
| Profesor: Željko Lukač | Student: Stefan Jovanović, RA234-2013 |
|  |  |
| Novi Sad, Jun 2018. | |

# Izveštaj

# Zadatak 1 (*Sample and Hold* algoritam):

Sample and Hold algoritam je najjednostavniji algoritam u kojem se za interpoliranu vrednost uzima poznata vrednost iz najbliže tačke.



Slika 1 SH Interpolacija

***Algoritam za traženje najbliže tačke***

Ovaj algoritam je i najbrži i zahteva najmanje procesorskih resursa, ali veliki nedostatak je stepeničasta struktura ivica, i neprirodno uniformisana tekstura slike (blokovska mustra).

**Rezultat:**



Slika 2 Slika pre obrade



Slika 3 Slika posle SH interpolacije, uvećanje 1.801

# Zadatak 2 (Bilinearna interpolacija):

Bilinearna interpolacija je malo složeniji algoritam u kojem se interpolirana vrednost računa usrednjavanjem vrednosti 4 piksela koji okružuju taj nepoznati piksel po sledećoj formuli:

Y = (1 – a) \* (1 – b) \* X(m,n) + (1 – a) \* b \* X(m + 1, n) + a \* (1 – b) \* X(m, n+1) + a \* b \* X(m + 1, n + 1)

Vrednosti a i b se računaju po formuli:

a = n / Sh – floor (n / Sh)

b = m / Sv – floor (m / Sv)

Gde su:

n, m – horizontalni/vertikalni index piksela u skaliranoj slici

Sh/Sv – horizontalni/vertikalni faktor skaliranja



Slika 4 Bilinearna interpolacija

Ova interpolacija zahteva malo više procesorskih resursa, ali daje puno bolje rezlutate, posebno sa povećanjem kvadrata interpolacije.

**Rezultat:**



Slika 5 Slika pre obrade



Slika 6 Slika posle bilinearne interpolacije, povećanje 1.801

# Zadatak 3 (Bikubična interpolacija):

Bikubična interpolacija je mnogo složeniji algoritam u kojem se interpolirana vrednost računa 16 piksela koji okružuju taj nepoznati piksel, ali tako da pikseli koji su bliži nepoznatom pikselu imaju veći uticaj. Kod bikubične interpolacije radi se kubična interpolacija po jednoj dimenziji, a zatim po drugoj.

Kubična interpolacija je predstavljena sledećom formulom:

Y(n) =

w(d) =



Slika 7 Bikubična interpolacija

Ova interpolacija zahteva mnogo vise procesorskih resursa, ali je dosta bolja interpolacija detalja u odnosu na prethodne dve.

**Rezultat:**

****

Slika 8 Pre obrade



Slika 9 BIkubična interpolacija, povećanje 1.801





## **Zadatak 4 (Rotacija slike – SH interpolacija):**

Još jedna operacija kod obrade slike koja podrazumeva korištenje neke interpolacije jeste rotacija slike.

Rotacija slike oko proizvoljne tačke radi se pomoću sledećih formula:

X’ = X \* cos(θ) – Y \* sin(θ) – m \* cos(θ) + n \* sin(θ) + m

Y’ = Y \* cos(θ) + X \* sin(θ) – m \* sin(θ) – n \* cos(θ) + n

m, n – koordinatne tačke oko koje se slika rotira



Slika 10 Slika pre obrade



Slika 11 Slika rotirana za 234°

## **Zadatak 5 (Rotacija slike sa bilinearnom interpolacijom):**

Rotacija se vrši kao u četvrtom zadatku, samo u ovom slučaju uz upotrebu bilinearne interpolacije.



Slika 12 Slika pre obrade



Slika 13 Slika rotirana za 234° + bilinearna interpolacija

# Bonus zadatak

Pošto je zahtevano u zadatku da dimenzije izlazne slike ostanu iste nakon vršenja rotacije, deo slike je potrebno iseći kako bi mogla stati u originalne dimenzije. Tako da se sa tim gubi odredjen broj piksela.



Slika 14 Slika rotirana za 234°

Zbog gore navedenog, pri vraćanju slike u prvobitno stanje, nije moguće vratiti izgubljene piksele.



Slika 15 Slika rotirana za 126°, vraćena u prvobitno stanje



Slika 16 Slika rotirana za 234° + bilinearna interpolacija



Slika 17 Slika rotirana za 126° + bilinearna interpolacija, vraćena u prvobitno stanje