



Univerzitet u Sarajevu
Elektrotehnički fakultet u Sarajevu
Odsjek za računarstvo i informatiku



DEBLURRING IMAGES USING AUTOENCODERS

Vještačka inteligencija

Predmetni profesor:

vanr. prof. dr. Amila Akagić

Mentor:

Elvedin Smajić

Studenti:

Naida Hasović 19196

Alma Jusufbegović 19393

Lana Malinov 19135

Sarajevo, 2024/2025

Sadržaj

Opis problema.....	4
Definicija osnovnih pojmova i primjena.....	5
Kratki pregled postojećih dataset-ova.....	8
Pregled stanja u oblasti deblurring-a.....	9
Pregled pristupa i metoda.....	9
Obim postignutih rezultata.....	9
Potencijalni pravci za poboljšanje.....	10
Analiza odabranog dataset-a.....	12
Rad na modelu.....	14
Opis metode.....	14
Izabrane tehnologije.....	14
Formatiranje podataka.....	14
Prikaz koda:.....	14
Kreiranje autoekodera.....	15
Definicija enkodera:.....	15
Prikaz koda:.....	15
Definicija dekodera:.....	16
Prikaz koda:.....	16
Spajanje enkodera i dekodera.....	16
Prikaz koda:.....	17
Treniranje modela.....	17
Prikaz koda:.....	17
Opis korištenih metrika.....	18
MSE (Mean Squared Error).....	18
MAE (Mean Absolute Error).....	18
PSNR (Peak Signal-to-Noise Ratio).....	18
SSIM (Structural Similarity Index).....	19
Diskutovanje dobijenih rezultata.....	19
Pregled treniranja modela.....	19
Evaluacija rezultata.....	19
Upotreba sa klasičnim metrikama.....	20
Vizuelna evaluacija.....	20
Primjer 1:.....	20
Primjer 2:.....	21
Primjer 3:.....	21
Vizuelna analiza rezultata sa nepoznatim podacima.....	22
Primjer 1:.....	22
Primjer 2:.....	22
Primjer 3:.....	23

Ograničenja i prijedlozi za budući rad.....	23
Zaključak.....	24
Literatura.....	25

Opis problema

Zamućenost (*engl. blurring*) je čest i značajan problem u digitalnim slikama, uzrokovan od strane različitih faktora, uključujući pokret kamere ili objekta, nepravilno fokusiranje, vibracije i fizičkih ograničenja optičkog sistema senzora. Rezultat toga su slike koje gube oštrinu, detalje i kontrast što direktno utiče na njihovu upotrebljivost u brojnim aplikacijama. U oblastima kao što su medicinska dijagnostika, autonomna vožnja, sigurnosni nadzor, prepoznavanje lica i satelitsko snimanje, kvaliteta slike je od presudne važnosti, a zamućene slike mogu dovesti do pogrešnih interpretacija i donošenja odluka.

Problem koji se rješava u ovom projektnom zadatku je rekonstrukcija oštre slike iz zamućene verzije korištenjem autoenkodera – posebnog tipa neuronske mreže koji uči kompresiju i rekonstrukciju podataka. Zadatak pripada širem naučnom području, poznatom kao *image deblurring*, koje se klasifikuje kao *inverse problem*, gdje je cilj procijeniti izvorni (*oštar*) signal na osnovu degradirane (*zamućene*) verzije.

Ono što ovaj problem čini izazovnim jeste činjenica da je proces zamućenja često nelinearan i zavisi od brojnih nepoznatih faktora – kao što su brzina i pravac kretanja, dužina ekspozicije, oblik zamućenja (tzv. *blur kernel*) i prisustvo šuma. Klasične metode dekonvolucije¹ često nisu dovoljno kompaktne i zahtijevaju eksplicitno modeliranje *kernelsa zamućenja*, što nije uvijek praktično u realnim uslovima.

Razvojem metoda dubokog učenja, naročito *konvolucionih mreža* i *autoenkodera*, otvorena je mogućnost da se pristupi ovom problemu na potpuno drugačiji način – putem učenja odnosa između zamućenih i oštarih slika direktno iz podataka, bez potrebe za ručnim definisanjem pravila ili *kernelsa*. Time se omogućava veća fleksibilnost i bolje sposobnosti modela, posebno kada se trenira na velikim skupovima podataka.

S obzirom na sve veću prisutnost slika sa zamućenjem (*blurring-om*) u realnim aplikacijama (*kamere, mobilni telefoni, dronovi, sateliti*), efikasno rješenje za deblurring ima širok spektar primjena i predstavlja važan segment unutar oblasti vještačke inteligencije i obrade slika.

¹ Matematički postupak suprotan konvoluciji u kojem se određuje jedna od podintegralnih funkcija kada su poznate druga podintegralna funkcija i rezultatna funkcija

Definicija osnovnih pojmova i primjena

Konvolucija (engl. *convolution*) je osnovna matematička operacija koja se koristi u obradi signala i slika, posebno u konvolucionim neuronskim mrežama (*CNN*). U kontekstu digitalnih slika, konvolucija predstavlja način na koji se informacija iz okoline svakog piksela koristi da bi se proizvela nova (*filtrirana ili transformisana*) verzija slike.

U najjednostavnijem smislu, konvolucija slike sa kernelom (*ili filterom*) se vrši tako što se mala matrica (npr. 3×3 ili 5×5 *kernel*) pomjera preko cijele slike. Na svakoj poziciji se vrši element-po-element množenje i sabiranje vrijednosti iz slike i kernela, čime se dobija novi izlazni piksel. Ova operacija omogućava detekciju ivica, zamagljivanje, izoštravanje, ekstrakciju karakteristika i dr.

Matematička definicija:

Za 2D diskretnu konvoluciju slike (*I*) i kernela (*K*):

$$S(x,y) = \sum_{i=-m}^m \sum_{j=-n}^n K(i,j) * I(x-i, y-j)$$

gdje je:

- $S(x, y)$ – nova vrijednost piksela na poziciji (x,y) ,
- $I(x, y)$ – originalna slika,
- $K(i, j)$ – vrijednosti kernela (filtera),
- m i n – polovine širine i visine kernela.

Primjena:

U problemu zamućenja slike, pretpostavlja se da je zamućena slika rezultat konvolucije oštre slike sa kernelom zamućenja (*koji može predstavljati pomjeranje kamere, defokus itd.*). Dakle, konvolucija se koristi za modeliranje procesa degradacije slike.

U deblurring-u, cilj je "**obrnuti konvoluciju**" – što je inverzni problem i često nerješiv bez dodatnih informacija, zbog čega se koristi duboko učenje za aproksimaciju.

Autoenkoder (engl. *autoencoder*) je vrsta neuronske mreže koja se sastoji iz dvije komponente: *enkodera*, koji pretvara ulazne podatke u sažet, nižedimenzionalni oblik, i *dekodera*, koji

pokušava rekonstruisati originalni podatak iz tog latentnog prikaza. Cilj autoenkodera je naučiti efikasnu reprezentaciju podataka, a u kontekstu obrade slika može se koristiti za uklanjanje šuma, zamućenosti ili rekonstrukciju dijelova slike.

Deblurring označava proces uklanjanja zamućenosti sa slike s ciljem rekonstrukcije njene izvorne oštine i strukturalnih detalja. Tradicionalni pristupi ovom problemu uključuju metode bazirane na filterima, dekonvoluciji i optimizaciji, dok savremene metode koriste duboko učenje (posebno konvolucione neuronske mreže (*CNN*) i autoenkodere), kako bi model automatski naučio rekonstruisati informacije izgubljene zamućenjem.

Zamućenje slike se najčešće modelira kao:

$$B = I * K + N$$

gdje su:

- B – zamućena slika (*blurred image*),
- I – izvorna oštra slika (*latent image*),
- K – kernel zamućenja (*blur kernel*),
- $*$ – operator konvolucije,
- N – aditivni šum.

Zamućena slika nastaje kao rezultat konvolucije originalne slike sa kernelom zamućenja, uz prisustvo šuma. U praksi, često nije poznato ni K ni N , što čini ovaj problem neodređenim i teškim za rješavanje, posebno kada je stepen zamućenosti visok.

Iako rezolucija slike (*broj piksela*) nije direktna mjera zamućenosti, niska rezolucija dodatno otežava rekonstrukciju slike jer sadrži manje informacija koje se mogu "izvući" iz slike.

Zamućenost se može kvantitativno procijeniti pomoću mjernih metrika kao što su:

- **MTF (*Modulation Transfer Function*)** – pokazuje koliko dobro sistem zadržava kontrast pri različitim prostornim frekvencijama.
- **PSF (*Point Spread Function*)** – opisuje kako tačka svjetlosti postaje raspoređena ("razmazana") u procesu snimanja.

- **IGM (*Image Gradient Magnitude*)** – oštre slike imaju izraženije gradijente (naglu promjenu vrijednosti između susjednih piksela), dok su zamućene slike glatke, s postepenim promjenama bez naglih prelaza.

Cilj deblurring modela je, dakle, da iz poznatog B nauči aproksimirati izvorni I , bez poznavanja eksplicitnog kernela K – što se u neuronaskim mrežama postiže učenjem nelinearne funkcije rekonstrukcije putem velikog broja primjera.

Primjena:

- **Medicina:** Poboljšanje kvaliteta snimaka poput MRI, CT i rendgenskih slika radi tačnije dijagnoze.
- **Sigurnosne kamere i nadzor:** Rekonstrukcija jasnih slika iz pokretnih i zamućenih video zapisa.
- **Autonomna vožnja:** Poboljšanje ulaznih podataka iz kamera za pouzdanije donošenje odluka.
- **Satelitske i zračne snimke:** Obrada slika zahvaćenih atmosferskim smetnjama.
- **Umjetnost i restauracija:** Obnavljanje starih ili oštećenih fotografija.

Kratki pregled postojećih dataset-ova

Za treniranje i evaluaciju modela za deblurring slika, dostupni su brojni javni skupovi podataka. Neki od najpoznatijih su:

- **GoPro Dataset**

Jedan od najkorištenijih datasetova za image deblurring. Sadrži parove zamućenih i oštarih slika snimljenih GoPro kamerama u pokretu. Koristi se u mnogim istraživanjima dubokog učenja za treniranje modela za *motion blur* uklanjanje.

- **REDS Dataset (*REalistic and Dynamic Scenes*)**

Dataset koji je dio NTIRE takmičenja. Fokusiran je na realistične video scene i uključuje veliki broj high-resolution slika i video sekvenci sa kontrolisanim zamućenjem.

- **Köhler Dataset**

Sadrži slike sa simuliranim zamućenjima pomoću poznatih *blur kernels*. Pruža visoku kontrolu nad tipom degradacije, korisno za testiranje algoritama u specifičnim uslovima.

- **ImageNet + Synthetic Blur**

ImageNet se često koristi kao osnovni dataset kojem se dodatno sintetički dodaje zamućenje radi treniranja modela u kontrolisanim uslovima.

Pregled stanja u oblasti deblurring-a

Pregled pristupa i metoda

- **Tradicionalne metode i optimizacija**

Rane tehnike (*prethodne dubokom učenju*) koristile su modele blur kernel-a, dekonvoluciju i Bayes/regularizaciju putem *PSF* i *MTF* funkcija. Primjer su Shrinkage Fields, koji kombinuju statistički pristup i Fourier transformaciju kako bi se rekonstruisale oštre slike.

- **End-to-end duboke neuronske mreže**

Početne CNN-bazirane metode direktno rekonstruišu latentnu sliku bez eksplicitnog određivanja kernela. Više arhitektura – npr. multiscale, U-net, rekurentne mreže – generišu visoke rezultate u terminima PSNR i SSIM.

- **Autoenkoderi (*kod-dekoder mreže*)**

Rade direktnu rekonstrukciju: primjer su dual autoenkoderi s separabilnim konvolucijama (denoising + deblurring) koji postižu visoku sličnost sa ground-truth slikama uz manje parametara. Nekoliko GitHub repozitorija (npr. *adi907/ImageDeblurring*, *sovit-123*, *rabia174*) demonstriraju upotrebu autoenkodera na delta-datasetu iz Kaggle (*Blur Dataset*) s PSNR/SSIM evaluacijom.

- **Hybrid rješenja (*kernel + net*)**

Model tipa „Analiza–Sinteza“ prvo pokušava da procijeni kako je slika zamućena (kernel), a zatim koristi te informacije da je izoštri. Ovakav pristup daje bolje rezultate u težim slučajevima zamućenja (*“uniform & 3D blur“*).

- **Difuzioni modeli i Transformeri**

Najnoviji radovi koriste diffusion modele i transformere za restoraciju slike. Swintormer koristi latentni diffusion model za generisanje karakteristika i Transformer blok za efikasnu rekonstrukciju. Razvoj MAE-stil pristupa sa proxy-deblurringom na ultrazvučnim slikama također pokazuje potencijal u specifičnoj domeni.

Obim postignutih rezultata

- **Benchmark vrijednosti PSNR/SSIM**

Metaanalize pokazuju da moderne deep-learning arhitekture (npr. *multiscale SRN*, *DMPHN*, *DeblurGAN-V2*) postižu PSNR >30 dB i SSIM >0.95 na datasetima poput

GoPro i Kohler.

- **Autoenkodori u praksi**

Dual-autoencoder i obični AE modeli na GitHub-u postižu preciznost (*accuracy*) između ~68–72% na Blur dataset-u, što potvrđuje mogućnost solidne rekonstrukcije.

- **Kerinsko modeliranje + net**

Analysis-Synthesis arhitektura pokazuje superiornost nad end-to-end CNN-ovima u blind slučajevima.

- **Najsavremeniji pravci**

Diffusion + Transformer hibridi kao Swintormer donose značajna poboljšanja i ostavljaju mjesto za istraživanje u efikasnosti i generalizaciji.

Potencijalni pravci za poboljšanje

- **Kombinacija modeliranja blur-kernela i AE-a**

Pokušaj integracije modula za procjenu blur-kernela unutar autoenkodera, slično "*analysis-synthesis*" mreži.

- **Uvođenje attention/transformer elemenata**

Trendovi iz Swintormer-a ukazuju da globalna perceptivna sposobnost transformera može poboljšati rekonstrukciju detalja.

- **Primjena diffusion modela**

Upotreba diffusion modela kao prior-generacijskog modula može povećati kvalitet uklanjanja zamućenja na stvarnim slikama s visokim nivoom šuma.

- **Specijalizacija za domene**

Analogno ultrazvučnim slikama, moguće je razviti AE-dodatke za specifične domene (npr. *medicinske snimke, tekstualne slike, lica*) uz korištenje semantičkih ili strukturalnih ograničenja.

- **Umanjenje broja parametara i ubrzanje**

Razdvojene (*separable*) konvolucije pokazuju veliki potencijal za brže i efikasnije modele što je idealno za aplikacije u realnom vremenu.

Zaključak:

Oblast uklanjanja zamućenja (*image deblurring*) je značajno napredovala prelaskom sa tradicionalnih metoda zasnovanih na kernel-estimaciji na end-to-end duboke neuronske mreže, posebno autoenkodere i konvolucione arhitekture. Ove metode danas postižu visok kvalitet rekonstrukcije. Autoenkoderi se ističu jednostavnošću i efikasnošću, dok napredniji hibridni pristupi - koji integrišu kernel modeliranje i neuronske mreže - omogućavaju robusniju obradu kompleksnih scenarija zamućenja.

Savremeni pravci razvoja, kao što su difuzioni modeli i transformeri, dodatno proširuju mogućnosti rekonstrukcije detalja i generalizacije na različite tipove slika. Integracija attention mehanizama i domensko prilagođavanje mreža (npr. za *medicinske slike*) nude nove potencijale za primjenu i istraživanje. Istovremeno, tehnike za redukciju kompleksnosti modela, poput separabilnih konvolucija, čine ove pristupe pogodnim za real-time upotrebu.

Zaključno, spoj tradicionalnih principa (*kao što je kernel estimacija*) sa savremenim modelima dubokog učenja pruža najperspektivniji pravac za buduće unapređenje performansi i efikasnosti image deblurring sistema.

Analiza odabranog dataset-a

Dataset koji je korišten za rješavanje ovog problema je Blur Dataset.

Ovaj dataset je objavljen na Kaggle-u (zauzima oko *170MB*) i dostupan je i preko [GitHub stranice](#), kao i u punoj rezoluciji na [Google Drive linku](#) (tačna veličina zavisi od preuzimanja, ali zauzima do nekoliko stotina MB). Može se preuzeti u smanjenoj rezoluciji (*do 2048 px po širini*) sa Kaggle-a, a originalne slike (u *.jpg* format) se mogu preuzeti u punoj rezoluciji sa Google Drive-a.

Dataset sadrži ukupno 1050 slika, organizovanih u 350 tripleta (svaki triplet sadrži 3 slike istog prizora: *sharp, defocused-blurred, motion-blurred*). Kako se sastoji isključivo iz slika, Blur Dataset nema tabličnih atributa, osim onih koji se mogu izvući iz imena samih fajlova. Kada je u pitanju klasifikacija, postoje ukupno 3 klase (*prema tipu zamućivanja*), i to:

- **S** – Sharp (*oštra slika*)
- **F** – deFocused-blurred (*zamućena zbog lošeg fokusa*)
- **M** – Motion-blurred (*zamućena zbog pokreta*)

Svaka od tri navedene klase ima tačno 350 slika, s tim da dataset nije unaprijed podijeljen na treniranje, validaciju i testiranje, već se to mora obaviti ručno po željenim razmjerama.

Rizici pri korištenju ovog dataset-a su:

- **Sličnost između klasa** - nefokusirane i motion-blurred slike ponekad mogu izgledati vrlo slično, što može zbuniti model ako se koristi za klasifikaciju ili evaluaciju.
- **Nepostojanje piksel-po-piksel podudaranja** - oštra i zamućena slika nisu snimljene identično (*nema savršenog poravnanja*), pa PSNR i SSIM nisu pouzdane metrike, što predstavlja izazov kod automatske evaluacije modela.
- **Veličina dataset-a** - veličina odabranog dataset-a, odnosno 1050 slika, može biti nedovoljno za treniranje kompleksnog autoenkodera.

U procesu rada sa autoenkoderima sa ciljem deblurring-a, pretprocesiranje slika je ključni korak koji omogućava efikasno i stabilno treniranje modela. Pretprocesiranje je od velike važnosti jer povećava efikasnost treniranja, stabilnost modela, i pomaže kod generalizacije, a neuronske mreže ne mogu raditi direktno sa sirovim slikama različitih dimenzija i raspona vrijednosti.

Najčešće metode pretprocesiranja su:

- **Promjena dimenzija (*resize*)** - sve slike se skaliraju na fiksnu dimenziju (npr. 128×128 ili 256×256 piksela) radi ujednačenosti ulaza u neuronsku mrežu.

- **Normalizacija piksela** - vrijednosti piksela se skaliraju iz opsega $[0, 255]$ u $[0, 1]$ (ili $[-1, 1]$) radi brže i stabilnije konvergencije mreže.
- **Uparivanje ulaz/izlaz** - za treniranje autoenkodera, svaka zamućena slika (*ulaz*) mora imati svoju oštru verziju (*izlaz*), tako da model uči da rekonstruiše čistu sliku iz zamućene.
- **Data augmentation (*opcionalna metoda*)** - rotacije, horizontalni flip, slučajni crop, koristi se da se model ispravno trenira i generalizuje bolje izlaze, pogotovo ako se radi sa dataset-om majeg opsega.

Rad na modelu

Opis metode

Za korištenje autoenkodera u modelu potrebno je kreirati dvije cjeline, prva je enkoder koji komprimira podatke, a druga je dekodek koji rekonstruiše sliku. Prednost korištenja ove metode je da se omogući mreži da nauči reprezentaciju slike i kako ukloniti šum/zamućenje bez potrebe za ručnim podešavanjem filtera.

Izabrane tehnologije

Korištene tehnologije uključuju:

- **Python** - jezik za razvoj i obradu podataka
- **TensorFlow/Keras** - za implementaciju i treniranje autoenkodera
- **NumPy** - za rad sa slikama i vektorizaciju podataka
- **Matplotlib** - za vizualizaciju ulaznih i izlaznih slika
- **Streamlit** - za izradu aplikacije koja korisniku omogućava da učitava zamućenu sliku i dobije rekonstrukciju.²

Formatiranje podataka

Svaka slika mora biti oblika (128, 128, 3) - širina, visina i RGB kanali, normalizovana na opseg piksela [0, 1].

Prikaz koda:

```
good_frames = 'blur-dataset/sharp'
bad_frames = 'blur-dataset/defocused_blurred'

clean_frames = []
for file in tqdm(sorted(os.listdir(good_frames))):
    if any(extension in file for extension in ['.jpg', 'jpeg', '.png']):
```

² Kod aplikacije se nalazi na *GitHub* repozitoriju: <https://github.com/lanamal/deblurify>.

```

        image = tf.keras.preprocessing.image.load_img(good_frames + '/' +
file, target_size=(128,128))
        image =
tf.keras.preprocessing.image.img_to_array(image).astype('float32') / 255
        clean_frames.append(image)

clean_frames = np.array(clean_frames)

blurry_frames = []
for file in tqdm(sorted(os.listdir(bad_frames))):
    if any(extension in file for extension in ['.jpg', 'jpeg', '.png']):
        image = tf.keras.preprocessing.image.load_img(bad_frames + '/' +
file, target_size=(128,128))
        image =
tf.keras.preprocessing.image.img_to_array(image).astype('float32') / 255
        blurry_frames.append(image)

blurry_frames = np.array(blurry_frames)

```

Kreiranje autoekodera

Definicija enkodera:

Enkoder se sastoji od tri konvolucijska sloja, gdje se za svaki sloj broj filtera povećava, a prostorna dimenzija slike smanjuje (*strides* = 2). Nakon konvolucijskih slojeva, izlaz se spljošti i proslijeđuje u potpuno povezani (*Dense*) sloj koji formira latentni vektor dimenzije 256. Cilj je kompresija ulazne slike u manju, reprezentativnu formu koja se koristi kao ulaz u dekode

Prikaz koda:

```

layer_filters = [64, 128, 256]
inputs = Input(shape = input_shape, name = 'encoder_input')
x = inputs

for filters in layer_filters:
    x = Conv2D(filters=filters,
        kernel_size=kernel_size,
        strides=2,
        activation='relu',
        padding='same')(x)

```

```

shape = K.int_shape(x)
x = Flatten()(x)
latent = Dense(latent_dim, name='latent_vector')(x)
encoder = Model(inputs, latent, name='encoder')

```

Definicija dekodera:

Dekoder prima latentni vektor iz enkoder dijela mreže, zatim ga proširuje u oblik pogodan za rekonstrukciju. Potom slijedi niz *Conv2DTranspose* (*dekonvolucionih*) slojeva koji postepeno povećavaju prostornu dimenziju slike (*upscaling*), uz korištenje ReLU aktivacione funkcije. U završnom sloju koristi se *Conv2DTranspose* sa 3 filtera (*za RGB kanale*) i *sigmoid* aktivacijom, čime se dobija rekonstruisana slika dimenzije (128, 128, 3) sa vrijednostima u rasponu [0, 1].

Prikaz koda:

```

latent_inputs = Input(shape=(latent_dim,), name='decoder_input')
x = Dense(shape[1]*shape[2]*shape[3])(latent_inputs)
x = Reshape((shape[1], shape[2], shape[3]))(x)
for filters in layer_filters[::-1]:
    x = Conv2DTranspose(filters=filters,
                        kernel_size=kernel_size,
                        strides=2,
                        activation='relu',
                        padding='same')(x)

outputs = Conv2DTranspose(filters=3,
                          kernel_size=kernel_size,
                          activation='sigmoid',
                          padding='same',
                          name='decoder_output')(x)

decoder = Model(latent_inputs, outputs, name='decoder')

```

Spajanje enkodera i dekodera

Spajanjem enkodera i dekodera finalno dobijamo traženi model - autoenkoder. Model se definiše tako da koristi isti ulaz kao enkoder, dok izlaz predstavlja rezultat dekodera nad izlazom enkodera.

Prikaz koda:

```
autoencoder = Model(inputs, decoder(encoder(inputs)), name='autoencoder')
```

Treniranje modela

Model se trenira nad ulaznim podacima (*mutne slike*) sa ciljem da nauči rekonstruisati odgovarajuće čiste slike. Validacija se vrši na istom skupu podataka. Tokom treniranja, parametri modela se optimizuju da minimiziraju MSE između originalnih i rekonstruisanih slika.

Treniranje je praćeno ReduceLROnPlateau callback-om kako bi se efikasnije podešavala brzina učenja.

ReduceLROnPlateau je Keras-ov callback koji automatski smanjuje stopu učenja (*learning rate*) kada model “*zapne*” – odnosno kada se vrijednost loss-a ili neke druge metrike ne poboljšava tokom više epoha.

Prikaz koda:

```
autoencoder.compile(
    loss='mse',
    optimizer='adam',
    metrics=[psnr_metric, ssim_metric, MeanAbsoluteError()])

lr_reducer = ReduceLROnPlateau(factor=np.sqrt(0.1),
                                cooldown=0,
                                patience=5,
                                verbose=1,
                                min_lr=0.5e-6)

callbacks = [lr_reducer]
history = autoencoder.fit(blurry_frames,
                          clean_frames,
                          validation_data=(blurry_frames, clean_frames),
                          epochs=100,
                          batch_size=batch_size,
                          callbacks=callbacks)
```

Opis korištenih metrika

MSE (*Mean Squared Error*)

MSE se koristi kao funkcija gubitka tokom treniranja autoenkodera. Ona mjeri prosječnu kvadratnu razliku između svake zamućene slike nakon rekonstrukcije i njene čiste (*ciljne*) verzije.

Manji MSE znači da je autoenkoder uspješno naučio kako da ukloni zamućenje i generiše slike slične originalnim.

Formula:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

MAE (*Mean Absolute Error*)

MAE je dodatna metrika kojom se mjeri prosječno odstupanje u pikselima između izlazne slike modela i ciljne slike.

Za razliku od MSE-a, MAE je manje osjetljiv na ekstremne vrijednosti i daje realniji uvid u prosječne greške modela, bez prenaglašavanja uticaja rijetkih velikih odstupanja.

Formula:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

PSNR (*Peak Signal-to-Noise Ratio*)

PSNR je važna metrika za mjerenje kvaliteta rekonstrukcije slike u kontekstu šuma i degradacije. U ovom projektu, PSNR mjeri koliko je "*rekonstruisana*" slika slična originalnoj čistoj slici u smislu intenziteta piksela.

Veća vrijednost PSNR-a ukazuje na veću sličnost i manje šuma, što znači bolji rezultat modela.

Formula:

$$PSNR = 10 \cdot \log_{10} \left(\frac{MAX^2}{MSE} \right)$$

Gdje je *MAX* maksimalna moguća vrijednost piksela.

SSIM (*Structural Similarity Index*)

SSIM je ključna metrika u ovom projektu jer bolje odražava perceptualnu sličnost između zamućene i rekonstruisane slike.

Dok MSE i PSNR gledaju razlike po pikselima, SSIM analizira strukturu, kontrast i svjetlinu, što je posebno važno u zadacima poput deblurring-a gdje je cilj očuvati teksture i detalje.

U rezultatima modela, vidljivo je da SSIM značajno raste nakon rekonstrukcije, što pokazuje da model ne samo da smanjuje greške po pikselima, već i poboljšava vizuelni kvalitet slike.

Zajedno, ove metrike omogućavaju kompletnu evaluaciju - numeričku i perceptualnu - uspješnosti autoenkodera u rješavanju zadatka uklanjanja zamućenja.

Diskutovanje dobijenih rezultata

Pregled treniranja modela

Model autoenkodera treniran je na skupu parova zamućenih i originalnih (*čistih*) slika, s ciljem da nauči proces rekonstrukcije slike kroz uklanjanje šuma i zamućenja. Kroz 100 epoha, model je pokazivao stabilan napredak – vrijednosti metrika na trening i validacionom skupu kontinuirano su se poboljšavale. Padajuća vrijednost MSE i MAE, uz rastuće vrijednosti PSNR i SSIM, sugerisu da je model uspješno minimizirao greške i poboljšao vizuelni kvalitet slika.

Evaluacija rezultata

Korištene metrike pokazuju sljedeće trendove:

- MSE se smanjuje tokom epoha, što potvrđuje smanjenje prosječne kvadratne greške između rekonstrukcije i ciljne slike.
- MAE, slična MSE, ali mjeri prosječnu apsolutnu grešku – otpornija je na ekstremne vrijednosti, također se smanjuje tokom epoha
- PSNR raste, što znači da se signal (*čista slika*) bolje razdvaja od šuma (*zamućenje*), te da je rekonstrukcija sve kvalitetnija.
- SSIM raste sa niske vrijednosti (*~0.28 prije obrade*) na znatno veću (*~0.83 nakon rekonstrukcije*), što potvrđuje da model ne samo da smanjuje šum, već i očuva strukturalne karakteristike slike.

Primjer:

- SSIM prije obrade: 0.2810
SSIM nakon obrade: 0.8714
Procentualno smanjenje šuma: +210.13%
- PSNR prije obrade: 14.06
PSNR nakon obrade: 29.67
Procentualno smanjenje šuma: +111.03%
- MAE prije obrade: 0.150438
MAE nakon obrade: 0.087128
Procentualno smanjenje greške: +42.08%

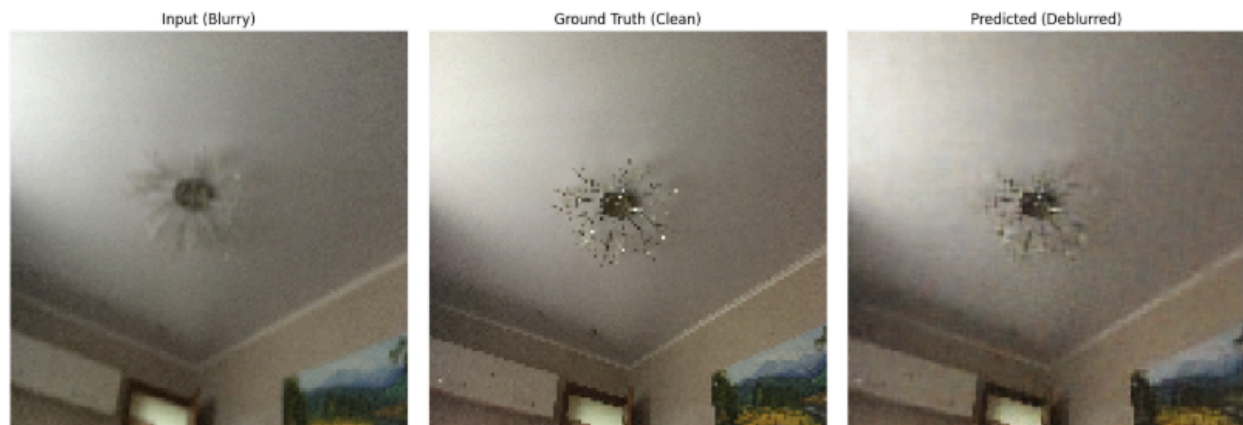
Upotreba sa klasičnim metrikama

U početku je korištena metrika **accuracy**, ali se pokazala neprikladnom za zadatak obrade slika. Accuracy mjeri tačnost klasifikacije, što podrazumijeva binarnu odluku po pikselu (da li je tačan ili ne). Accuracy ne uzima u obzir intenzitet greške između piksela, niti strukturalne karakteristike slike, što je ključno kod zadatka poput deblurring-a.

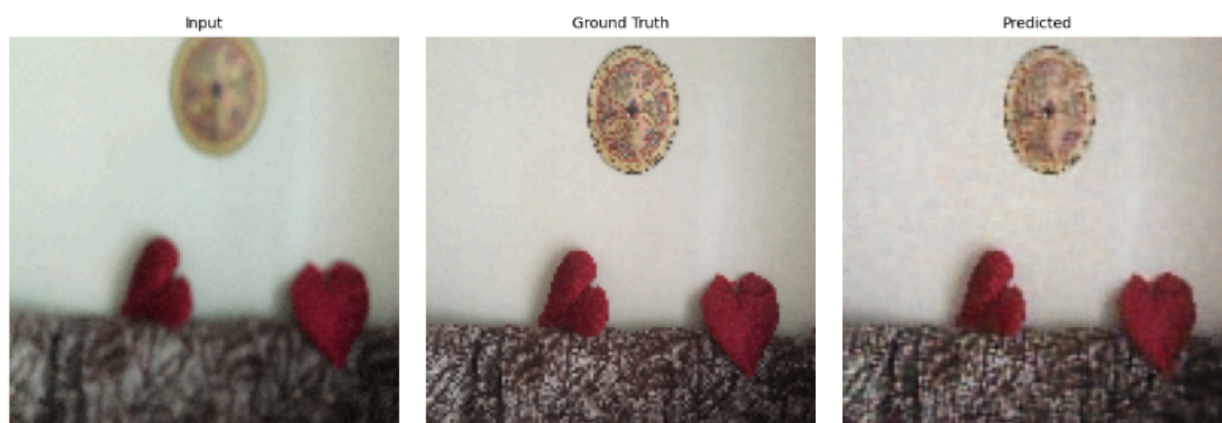
Nasuprot tome, PSNR i SSIM uzimaju u obzir kvalitet slike u širem smislu: percepciju korisnika, kontrast, teksture i strukturu. Stoga su ove metrike mnogo adekvatnije za evaluaciju kvaliteta rekonstrukcije slike.

Vizuelna evaluacija

Primjer 1:



Primjer 2:



Primjer 3:

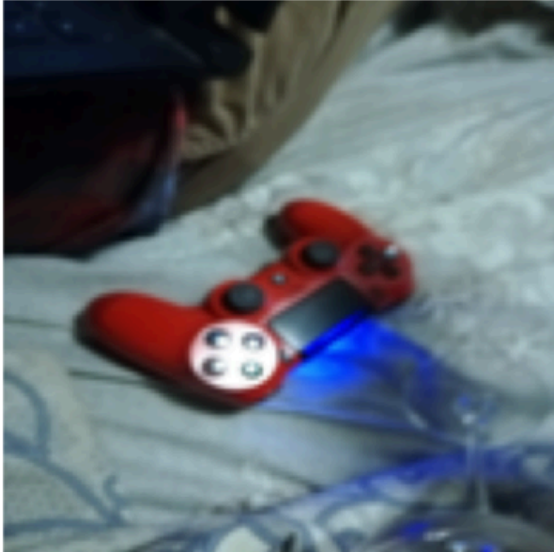


Vizuelna poređenja između ulaznih zamućenih slika, originala i rekonstruisanih rezultata potvrđuju numeričke nalaze, odnosno da model uspješno uklanja zamućenje, vraća detalje i poboljšava kontrast, naročito na rubovima objekata i teksturama.

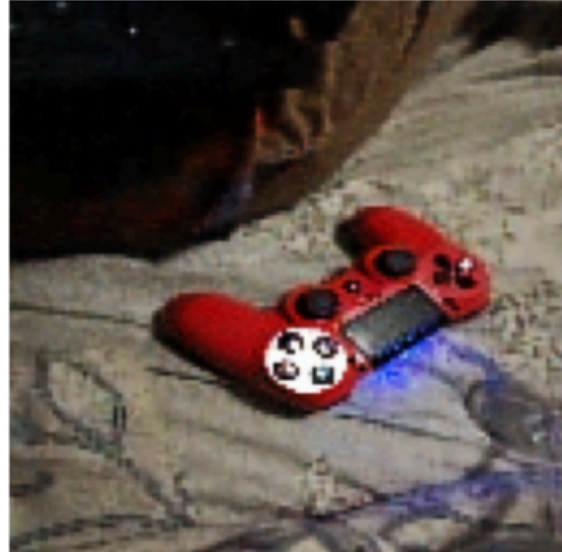
Vizuelna analiza rezultata sa nepoznatim podacima

Primjer 1:

Blurred Image



Deblurred Image



Primjer 2:

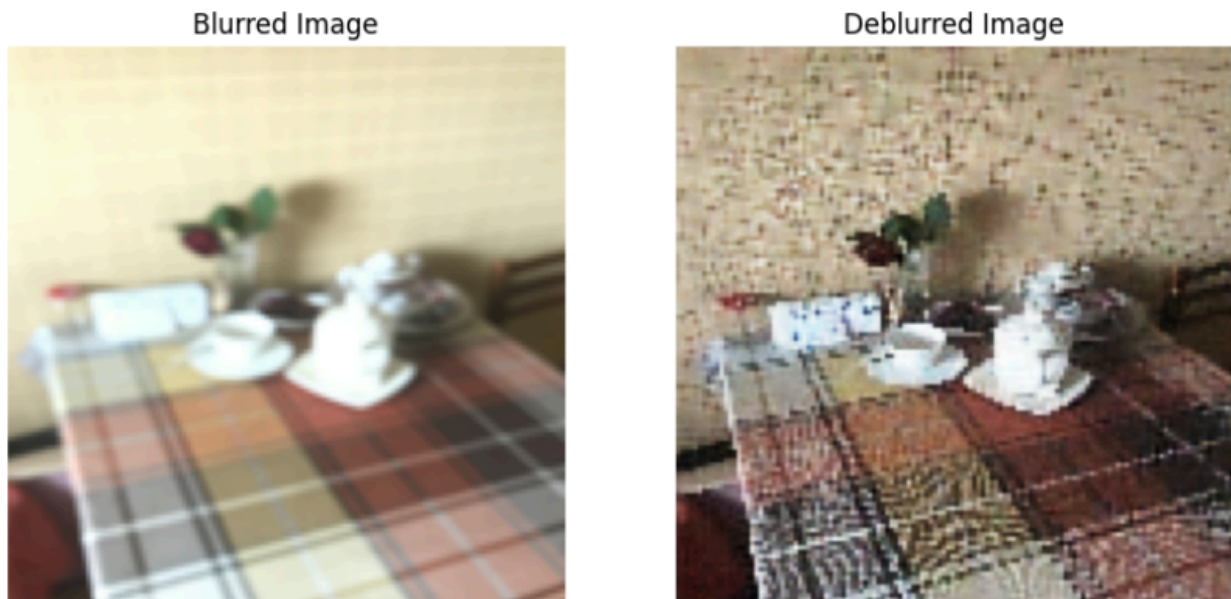
Blurred Image



Deblurred Image



Primjer 3:



Vizualizacija potvrđuje da model radi sasvim korektno sa ulazima na kojima nije treniran. Model je uspješno vratio oštrinu ključnih objekata, poboljšao kontrast i jasnoću slike i rekonstruisao teksturu i uzorke na pozadini. Iako je kvalitet slike značajno poboljšan, prisutan je blagi šum u rekonstruisanoj slici, što govori da model idalje ima prostora za poboljšanje.

Ograničenja i prijedlozi za budući rad

Iako su rezultati veoma dobri, u određenim slučajevima (npr. *vrlo zamućene slike*) model nije u potpunosti uspio vratiti sve detalje. Postoji i ograničenje za dimenzije slika, odnosno model ne vraća poželjne rezultate za slike većih dimenzija.

Za poboljšanje performansi u budućnosti mogu se razmotriti sljedeće strategije:

- Korištenje **perceptual loss funkcije** (npr. *VGG loss*), koja koristi karakteristike visoko-treniranih mreža za preciznije učenje strukture slike.
- Uvođenje **generativnih modela (GAN-ova)** koji mogu dodatno poboljšati vizuelnu realističnost rezultata.
- Kombinovanje sa **rezidualnim mrežama** koje bolje rješavaju problem prenosa finih detalja.
- Augmentacija podataka različitim vrstama šuma i zamućenja za generalizaciju modela.

Zaključak

U ovom radu je prikazan uspješno je implementiran model autoenkodera za zadatak deblurring-a slika, koristeći javno dostupan skup podataka koji sadrži trojke oštarih, defokusirano-zamućenih i zamućenih pokretom slika. Kroz faze pretprocesiranja podataka, model je treniran da rekonstruiše oštru sliku na osnovu zamućene, a postignuti rezultati pokazuju zadovoljavajuću sposobnost mreže da prepozna i ukloni različite vrste zamućenja.

U poređenju sa rezultatima iz prethodne faze (npr. *jednostavne konvolucijske klasifikacije zamućenih vs. oštarih slika*), autoenkoder je pokazao viši stepen sposobnosti u analizi i interpretaciji vizualnih obrazaca jer ne samo da klasifikuje, već i rekonstruiše informaciju. To predstavlja značajan korak naprijed ka praktičnoj upotrebi modela u realnim aplikacijama poput poboljšanja kvaliteta fotografija, videa i medicinskih slika.

Ipak, postoje određena ograničenja pri izvršavanju. Zbog toga što slike u dataset-u nisu savršeno poravnate (*nema piksel-po-piksel parova*), evaluacija kvaliteta rekonstrukcije pomoću metrika poput SSIM i PSNR nije sasvim pouzdana. Takođe, korišten je ograničen broj podataka, što može uticati na sposobnost generalizacije modela. Veće performanse bi vjerovatno bile postignute korištenjem naprednijih arhitektura poput U-Net, GAN-a ili uvođenjem mehanizama pažnje (*attention*), kao i dodatnim augmentacijama podataka.

Za budući rad preporučuje se korištenje većih i preciznije usklađenih skupova slika, isprobavanje naprednijih modela te poređenje rezultata sa tada postojećim rješenjima iz literature, kako bi se tačnije procijenila uspješnost ovog pristupa.

Literatura

1. Nah, S., Hyun Kim, T., & Mu Lee, K. (2017). Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
https://openaccess.thecvf.com/content_cvpr_2017/papers/Nah_Deep_Multi-Scale_Convolutional_CVPR_2017_paper.pdf
2. Nah, S., Baik, S., Hong, S., Moon, G., Son, S., Timofte, R., & Mu Lee, K. (2019). NTIRE 2019 Challenge on Video Deblurring and Super-Resolution: Dataset and Study. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.
https://openaccess.thecvf.com/content_CVPRW_2019/papers/NTIRE/Nah_NTIRE_2019_Challenge_on_Video_Deblurring_and_Super-Resolution_Dataset_and_CVPRW_2019_paper.pdf
3. Köhler, R., Hirsch, M., Mohler, B., Schölkopf, B., & Harmeling, S. (2012). Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. European Conference on Computer Vision (ECCV).
<https://webdav.tuebingen.mpg.de/pixel/benchmark4camerashake/>
4. <https://www.reddit.com/r/MachineLearning/comments/gdt35p>
5. https://en.wikipedia.org/wiki/Shrinkage_Fields_%28image_restoration%29
6. <https://arxiv.org/abs/1907.10128>
7. <https://github.com/adi907/ImageDeblurring>
8. <https://arxiv.org/abs/2308.09388>
9. <https://arxiv.org/abs/2004.02956>
10. <https://pubmed.ncbi.nlm.nih.gov/36135415/>
11. <https://arxiv.org/abs/2306.08249>
12. <https://link.springer.com/article/10.1007/s11042-023-17771-y>
13. <https://www.themoonlight.io/en/review/efficient-image-deblurring-networks-based-on-dif-fusion-models>
14. Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing* (4th ed.). Pearson.