

Image Colorization Project

Paweł Małecki

14.06.2024

I. Wprowadzenie

Projekt koncentruje się na kolorowaniu czarno-białych obrazów za pomocą architektur typu U-Net oraz autoencoderów. Do realizacji tego zadania wykorzystano zbiór CIFAR-10 oraz przestrzeń kolorów LAB.

II. Podjęte Kroki

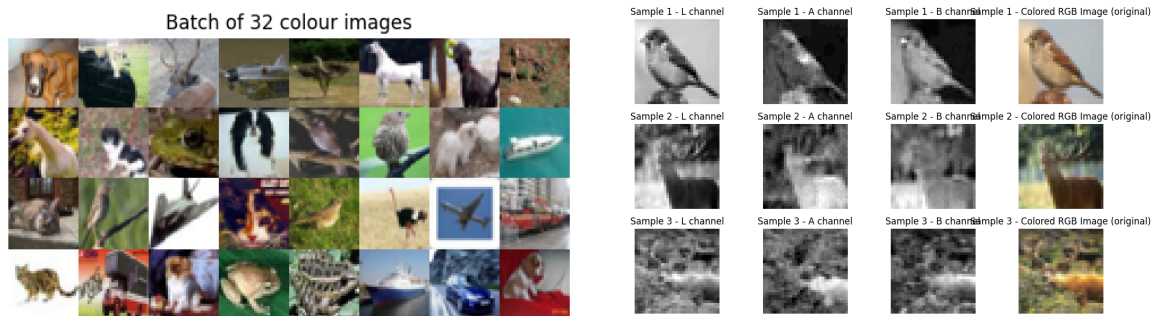
1. Podział danych, załadowanie dataloaderem, rzeksztalcenie do przestrzeni kolorów LAB i augmentacja.
2. Implementacja modeli i ich wariantów.
3. Trening, walidacja i ocena jakości pokolorowanych obrazów.

III. Metryki

Użyte zostały metryki MSE (Mean Squared Error) oraz SSIM (Structural Similarity Index) do oceny jakości wyników.

IV. Dane

Dane pochodziły z zestawu CIFAR-10, który zawiera różnorodne kolorowe obrazy. Dane zostały podzielone na zestawy treningowe, walidacyjne i testowe w proporcjach odpowiednio 80%, 10%, 10%. Zastosowano augmentację danych, w tym obrót, przesunięcia i odbicia lustrzane, aby zwiększyć różnorodność danych treningowych.



Obrazy CIFAR-10 zostały przekształcone z formatu RGB do przestrzeni barw LAB, gdzie kanał jasności (L) został użyty jako wejście modeli, a kanały AB zostały wykorzystane jako wyjście, co usprawniło proces uczenia modeli kolorowania. Połączenie kanału L z kanałami AB pozwoliło na uzyskanie kolorowych obrazów.

V. Modele

Wybór modeli opierał się na próbie różnorodnych architektur od prostego kodera-dekodera po bardziej zaawansowane jak U-Net, czy autoencodery z dodatkowymi warstwami jak BatchNormalization, różnymi rodzajami pooling'u i wielkością wektora ukrytego oraz sieci rekurencyjne RNN z różną ilością komórek LSTM.

VI. Wyniki

Poniżej przedstawiono kluczowe wyniki dla wybranych modeli:

Model (30 epochs)	Params (total)	Validation MSE	Validation SSIM	Validation Acc	Test MSE	Test SSIM	Test Acc
Simple Encoder-Decoder	23,162	0.0024	0.7748	0.6283	0.0024	0.7726	0.6347
U-Net with BatchNormalization and MaxPooling2D	3,104,322	0.0024	0.7669	0.6476	0.0025	0.7640	0.6485
U-Net with BatchNormalization and AveragePooling2D	3,104,322	0.0025	0.7647	0.6451	0.0025	0.7625	0.6498
U-Net with Skip Connections and MaxPooling2D	31,030,658	0.0025	0.7817	0.6490	0.0026	0.7777	0.6486
Autoencoder with latent dimension 64	1,858,178	0.0027	0.7665	0.6333	0.0028	0.7648	0.6349
Autoencoder with latent dimension 256	3,431,234	0.0028	0.7614	0.6254	0.0028	0.7599	0.6310
Autoencoder with latent dimension 256 and BatchNormalization	3,434,818	0.0027	0.7687	0.6268	0.0028	0.7649	0.6272
Autoencoder with VGG16 Backbone	25,966,210	0.0024	-	0.6392	0.0025	-	0.6440
RNN (LSTM 128)	854,528	0.0026	0.7767	0.6100	0.0026	0.7743	0.6204
RNN (LSTM 256)	1,838,080	0.0025	0.6095	0.7788	0.0026	0.7741	0.6207



VII. Dalsze kroki

Osiągnięte wyniki są obiecujące, szczególnie dla modeli opartych na architekturze U-Net oraz zaawansowanych wersjach autoencoderów. Możliwe dalsze kroki obejmują:

- Fine-tuning hiperparametrów dla poprawy skuteczności i stabilności modeli.
- Implementacja innych modeli: VAE z CNN, CGAN, WGAN.
- Implementacja modeli fusion do łączenia informacji z różnych modeli kolorowania.
- Zmiana datasetu na obrazy o wyższej rozdzielczości, np. 256x256 pikseli.
- Po kolorowaniu obrazów można zastosować techniki super-rozdzielczości (super-resolution).