Dekompresja ścieżek audio wypowiedzi głosowych z wykorzystaniem transferu stylu

Maciej Gruszczyński
228131

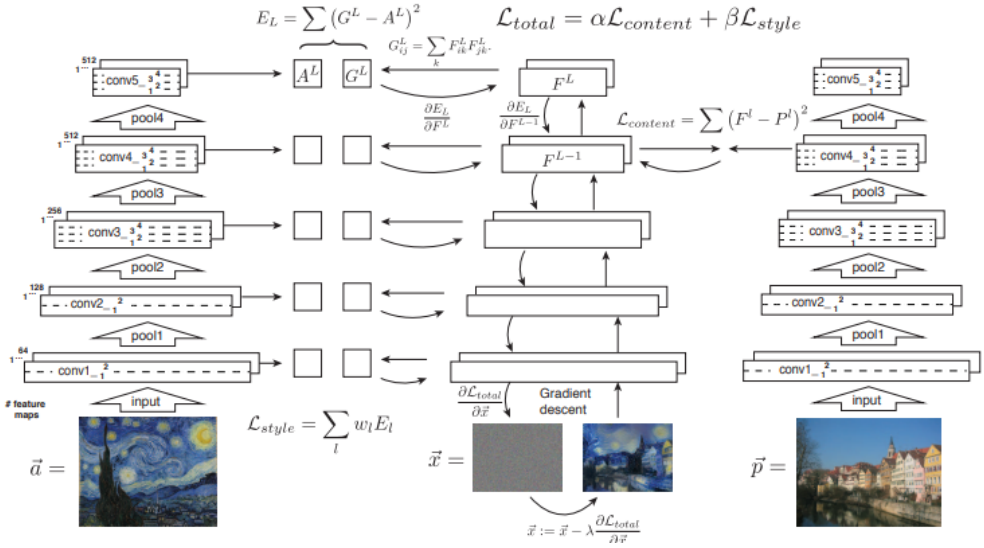
Data oddania: 28.01.2020

Abstract

Dokument jest raportem z badań nad problemem dekompresji ścieżek audio wypowiedzi głosowych z wykorzystaniem dostępnych modeli transferu stylu, przeprowadzonych w ramach przedmiotu "Głębokie modele uczenia maszynowego".

1 Przegląd literatury

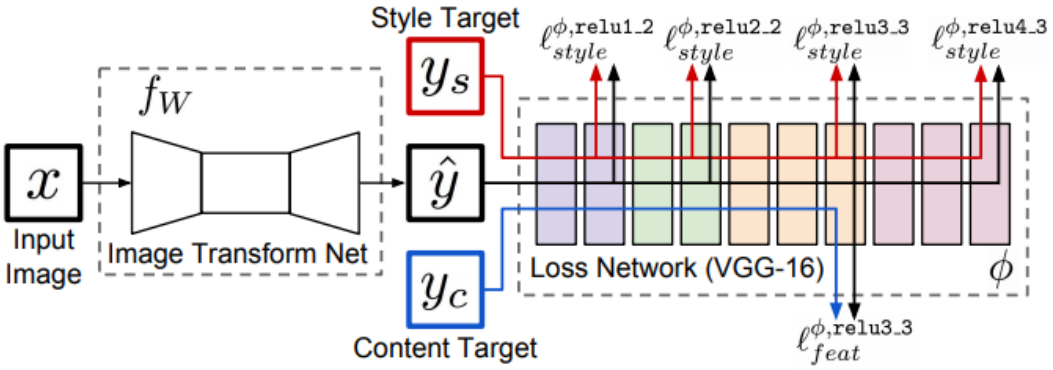
Można wyróżnić dwie dominujące architektury w transferu stylu zarówno obrazu jak i dźwięku. Pierwsza to sieć konwolucyjna [1], która na wejściu przyjmuje obraz będący szumem i metodą gradientową zmienia obraz tak, aby posiadał zadany styl i treść. Docelowe styl i treść są natomiast wartościami neuronów wybranej pretrenowanej sieci (np. VGG dla obrazów) w wybranych warstwach dla dwóch różnych obrazów będących kolejno źródłem treści i stylu. Funkcja straty polega na liczeniu różnicy między wartościami w wybranych warstwach 3 sieci - treści, stylu i generowanego obrazu. Jest to pierwsza zaproponowana architektura i została ulepszona i zmodyfikowana w kilku pracach. Jej podstawową wadą jest długi czas generowania obrazu, wykluczający ją z użycia w systemach czasu rzeczywistego.



Rysunek 1: Architektura z publikacji [1], Gatys

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Drugi typ architektury to autokoder. Pierwszy raz zaproponowany przez Johnson et al. w publikacji [2]. Autokoder jest uczony tak, aby nakładać styl na różne podawane obrazy jednocześnie zachowując treść każdego z nich. Definicja stylu i treści oraz funkcja straty są takie same jak w poprzedniej architekturze. Różnica jest taka, że generowany obraz po wyjściu z autokodera musi przejść dodatkowo przez sieć ekstrahującą styl i treść, aby obliczyć błąd. Zaletą tego podejścia jest możliwość generowania obrazów o zmienionym stylu w czasie rzeczywistym w trybie feed-forward, pod warunkiem, że wcześniej wyuczyliśmy autokoder.

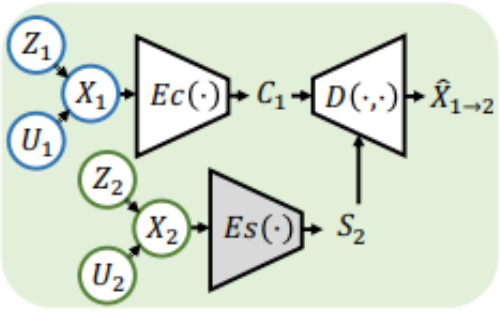


Rysunek 2: Architektura z publikacji [2], Johnson

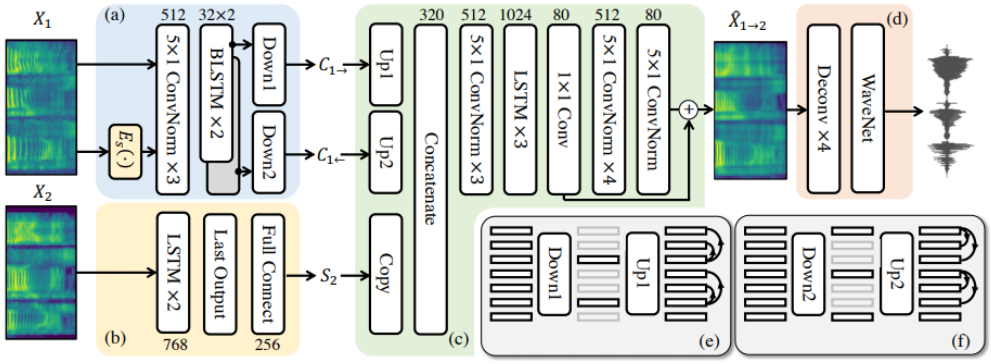
Powyższe architektury oraz ich modyfikacje można zastosować do dźwięku. Wystarczy przekształcić ścieżkę audio do postaci spektrogramu. Istnieje co najmniej kilkanaście publikacji i repozytoriów, w których można znaleźć różne próby dokonywania transferu stylu dźwięku bazujące na publikacjach odnoszących się do obrazu. Głównym problemem, który występuje w tym zagadnieniu jest wrażliwość ucha ludzkiego na wszelkie zaburzenia pojawiające się w spektrogramie, a następnie w przebiegu fali, czyli dźwięku. Większość modeli generuje mocno zaburzone ścieżki audio. Kolejnym problemem jest trudność w ocenie takich ścieżek. Najprostszym sposobem wydaje się być subiektywna ocena. Do badań wybrano model, który przez czteroosobową grupę został wybrany jako generujący dźwięk najlepszej jakości.

2 Pierwsza wizja rozwiązania

Pierwotnie za podstawę do badań miała posłużyć architektura AutoVC z publikacji [3]. Architektura składa się z kodera stylu i osobnego kodera treści oraz dekodera. Kodery dostarczają embedding stylu i treści, na podstawie których dekoder generuje wynikową wypowiedź.

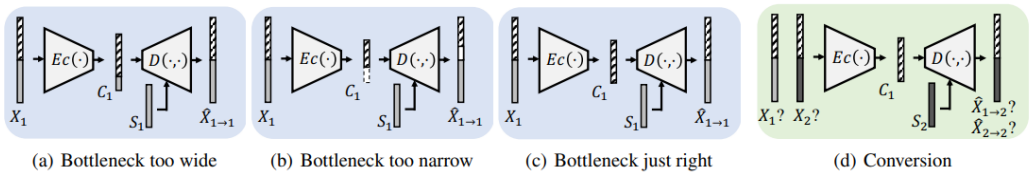


Rysunek 3: Architektura AutoVC na poziomie koderów i dekodera.



Rysunek 4: Architektura AutoVC na poziomie warstw.

Uczenie modelu rozpoczyna się od wyuczenia osobno kodera stylu (Es) zapożyczonego z publikacji [4], Wan et al. Następnie uczony jest cały autokoder. Jego zadaniem jest odtworzenie tej samej ścieżki audio na wyjściu co na wejściu. Nie jest to jednak problem nawet, gdybyśmy nie używali kodera stylu, ponieważ koder stylu pojawia się również jako składowa kodera treści. Więc opierając się na samym koderze treści możemy odtworzyć wejście na wyjściu. Aby zmusić model do wykorzystywania informacji o głosie zwartej w embeddingu dostarczanym przez koder stylu, stosuje się zabieg zmniejszania embeddingu treści, tak aby w trakcie uczenia zachować jakość odtwarzania wejścia na wyjściu i jednocześnie zachować w embeddingu treści wyłącznie informację o treści, a styl czerpać z embeddingu stylu. Obrazuje to poniższy obrazek.

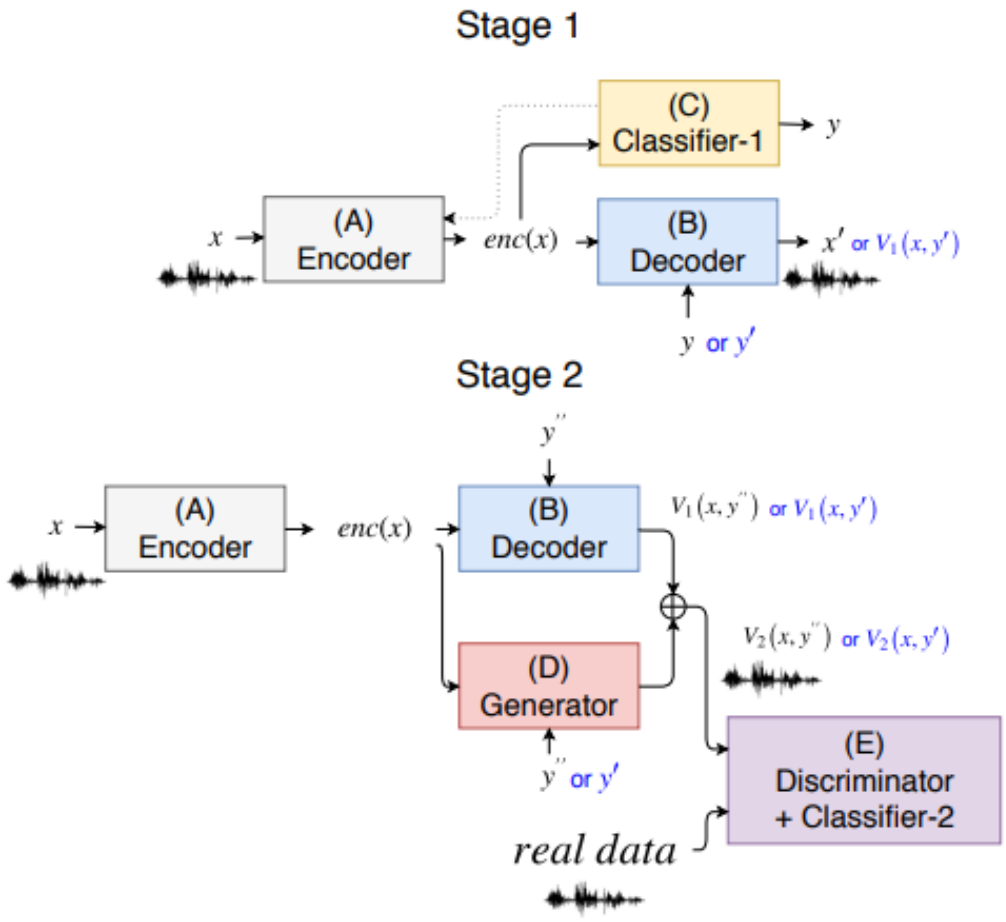


Rysunek 5: Schemat strojenia rozmiaru embeddingu treści.

W sieci dostępne jest publiczne repozytorium zawierające fragment kodu modelu AutoVC. Jednak ze względu na ukrywanie przez autora szczegółów takich jak parametry algorytmu generowania spektrogramów lub dokładny sposób rozpoznania właściwej długości embeddingu oraz nieznany czas uczenia, odtworzenie metody może być zbyt czasochłonne. Z tego powodu zaproponowano kolejny model.

3 Druga wizja rozwiązania

Wybrano model z publikacji [5], Chou et al., dla którego udostępniony jest pełny kod. Architekturę modelu można podzielić na dwie części. Część pierwsza składa się z autokodera, którego zadaniem jest rekonstrukcja spektrogramu. Embedding autokodera poddawany jest klasyfikacji na klasę jednego z kilku predefiniowanych mówców. Rezultat klasyfikacji wykorzystywany jest jako regularyzacja w trakcie uczenia autokodera, która ma zapewnić, że embedding nie będzie zawierał informacji o tonie głosu mówcy, a wyłącznie o treści wypowiedzi. Druga część to sieć GAN, która generuje dodatek do wyjściowego spektrogramu, który ma za zadanie sprawić, że wygenerowany spektrogram będzie wyglądał bardziej jak spektrogramy prawdziwych ścieżek audio. Dyskryminatorem jest sieć konwolucyjna, której zadaniem jest rozróżnienie czy spektrogram pochodzi z prawdziwej wypowiedzi, czy został wygenerowany.



Rysunek 6: Architektura z publikacji [5], Chou et al.

Celem tej pracy jest użycie opisanego modelu do odtwarzania oryginalnego nagrania wypowiedzi na wyjściu, gdy podamy nagranie skompresowane na wejściu. Praca polega na zbadaniu czy istnieje taka możliwość oraz jaki przynosi skutek.

4 Rekonstrukcja sygnału audio ze spektrogramu

W modelu AutoVC do rekonstrukcji sygnału audio ze spektrogramu wykorzystano model WaveNet. WaveNet jest stworzony przez firmę DeepMind oraz wykorzystywany obecnie w modelach text-to-speech firmy Google. Charakteryzuje się dłuższym czasem działania lecz lepszą jakością ścieżki dźwiękowej na wyjściu niż algorytm Griffin-Lim zastosowany w modelu z publikacji [5], Chou et al. Jednym z elementów badań jest wykorzystanie modelu WaveNet. Ponadto zbadano różnicę w wykorzystywaniu spektrogramu logarytmicznego (mel) oraz liniowego jako dwóch alternatywnych form reprezentacji ścieżek audio.

5 Źródła kodu, biblioteki

Repozytorium wykorzystanego modelu transferu stylu [5]:
https://github.com/jjery2243542/voice_conversion

Repozytorium WaveNet, z którego wykorzystano kod generowania mel-spektrogramu i rekonstrukcji ścieżki dźwiękowej:
https://github.com/r9y9/wavenet_vocoder

Model WaveNet z repozytorium AutoVC pretrenowany na zbiorze VCTK:
https://drive.google.com/file/d/1Zksy0ndlDezo9wclQNZYkGi_6i7zi4nQ/view

Zbiór wypowiedzi głosowyh VCTK:
https://homepages.inf.ed.ac.uk/jyamagis/page3/page58/page58.html

Wykorzystane biblioteki to: torch, librosa, lws i wavenet_vocoder.

6 Badania, wyniki

Do badań wykorzystano kartę graficzną RTX 2070. Czas uczenia pierwszej części modelu (autokodera) wynosi 6 godzin, natomiast drugiej części (GAN) 20 godzin, stąd bardzo ograniczona liczba prób.

Próbkowanie oryginalnej ścieżki to 48kbps, natomiast po kompresji 8kbps. W ten sposób skompresowano wszystkie nagrania, aby następnie nauczyć autokoder je dekompresować.

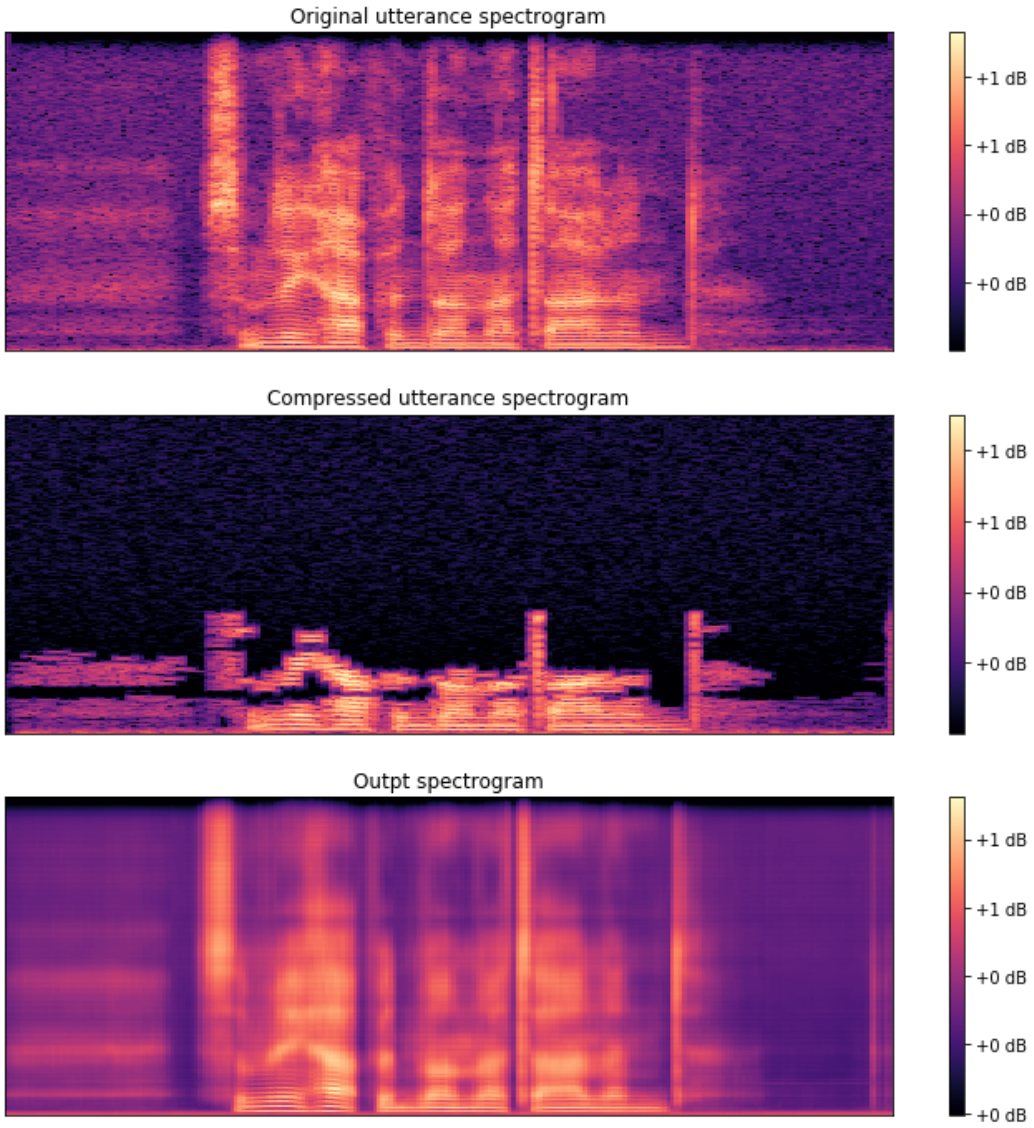
Jakość rekonstrukcji mierzona jest funkcją straty, czyli średnią z wartości bezwzględnych różnicy między spektrogramem wyjściowym i wzorcowym (ang. self reconstruction loss). Im wartość jest mniejsza tym lepsza rekonstrukcja spektrogramu. Modele są testowane na 20% całego zbioru VCTK nie biorących udziału w fazie uczenia.

Pogrubionym tekstem zaznaczono parametr zmieniający się w danej serii pomiarów przy stałych wartościach pozostałych parametrów.

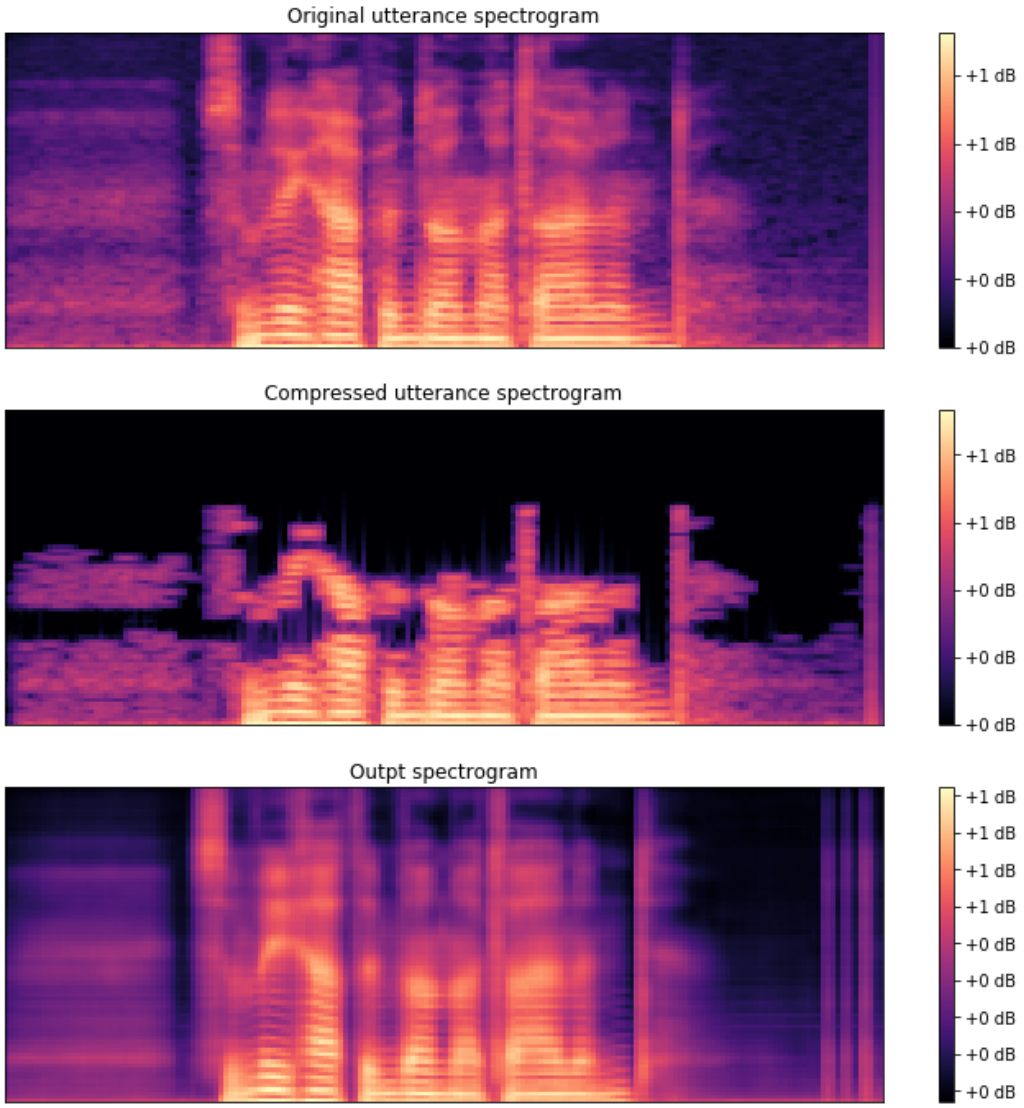
tryb	**typ spektrogarmów**	GAN	**rozmiar embeddingu**	**liczba epok**	**strata rekonstrukcji**
zamiana głosu	liniowy	nie	512	80000	1.950
dekompresja	liniowy	nie	512	80000	2.344
dekompresja	logarytmiczny	nie	512	**70000**	1.438
dekompresja	logarytmiczny	nie	512	**80000**	1.411
dekompresja	logarytmiczny	nie	512	**90000**	1.448
dekompresja	logarytmiczny	nie	**384**	80000	1.498
dekompresja	logarytmiczny	nie	**640**	80000	1.432
dekompresja	logarytmiczny	tak	512	**110000**	2.138
dekompresja	logarytmiczny	tak	512	**120000**	2.210
dekompresja	logarytmiczny	tak	512	**130000**	2.255
dekompresja	logarytmiczny	tak	512	**146000**	2.080
dekompresja	logarytmiczny	tak	512	**150000**	2.182
dekompresja	logarytmiczny	tak	512	**160000**	2.206
dekompresja	logarytmiczny	tak	512	**173000**	2.411

Tabela 1: Wyniki badań

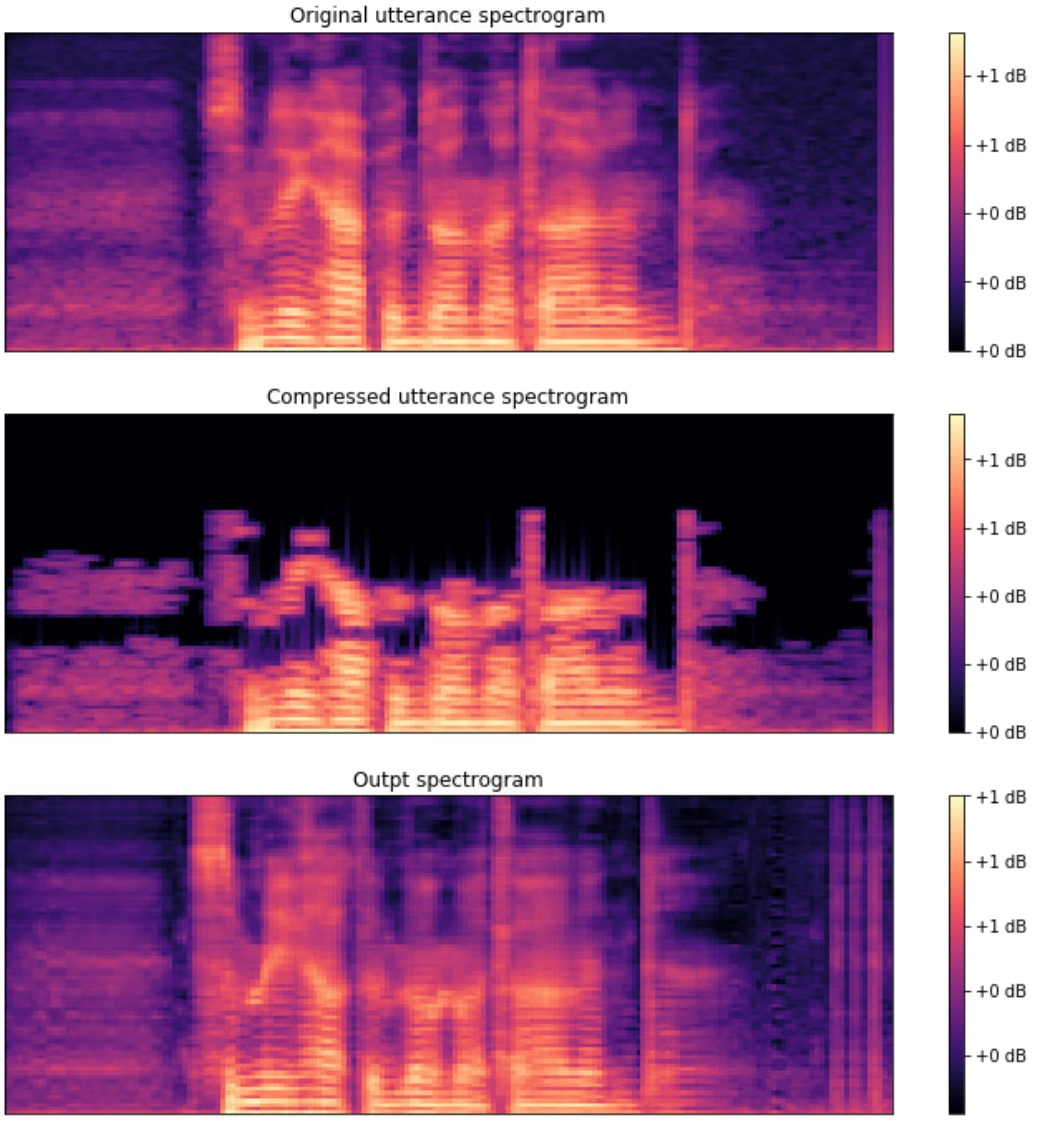
7 Reprezentacja graficzna



Rysunek 7: Trzy spektrogramy liniowe: przed kompresją, po i wygenerowany



Rysunek 8: Trzy mel-spektrogramy: przed kompresją, po i wygenerowany dla najlepszej znalezionej konfiguracji hiperparametrów bez modułu GAN



Rysunek 9: Trzy mel-spektrogramy: przed kompresją, po i wygenerowany dla najlepszej znalezionej liczby epok z modułem GAN

8 Podsumowanie

Wyniki wskazują, że najlepszą jakość dekompresji uzyskujemy, gdy model pracuje na mel-spektrogramach z embeddingiem o rozmiarze 512 i uczy autokoder przez 80 tysięcy epok. Jednak badanie średniej wartości rekonstrukcji spektrogramu jest oczywiście niezależne od użytej metody rekonstrukcji ścieżki dźwiękowej. W celu porównania algorytmu Griffin-Lim oraz modelu WaveNet należy zapoznać się z plikami audio w repozytorium projektowym w folderze *samples*. Porównano tam dekompresję spektrogramów liniowych oraz najlepszą logarytmicznych z wynikowej Tabeli 1 dla wypowiedzi mężczyzny i kobiety.

Bibliografia

[1] Gatys et al. (2016) *Image Style Transfer Using Convolutional Neural Networks*

[2] Johnson et al. (2016) *Perceptual Losses for Real-Time Style Transfer and Super-Resolution*

[3] Qian et al. (2019) *AUTOVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss*

[4] Wan et al. (2018) *Generalized End-to-End Loss for Speaker Verification*

[5] Chou et al. (2018) *Multi-target Voice Conversion without Parallel Data by Adversarially Learning Disentangled Audio Representations*