Final Technical Report

St. Nisrina Nabila, Faradias Izza A. F. dan Nurul Salsabila S.
Program Studi Sistem Informasi, Departemen Matematika
Universitas Hasanudddin

CONTENTS

LIST OF FIGURES

Final Technical Report

***Abstract*—*Image to image translation* adalah proses transformasi gambar dari satu domain ke domain lain, di mana tujuannya adalah untuk mempelajari pemetaan antara gambar input dan gambar output. Tugas ini umumnya dilakukan dengan menggunakan satu set pelatihan dari pasangan gambar yang diselaraskan. Sebagai model *image to image translation* Pix2Pix secara langsung dapat diterapkan untuk tugas-tugas segmentasi gambar. Keuntungan arsitektur pix2pix adalah bersifat generik dan mempelajari tujuan selama pelatihan tanpa membuat asumsi apa pun antara dua jenis gambar. Oleh karena itu, model ini fleksibel untuk berbagai situasi.**

I. Pendahuluan

Generative Adversarial Networks, atau disingkat GAN, adalah pendekatan untuk pemodelan generatif menggunakan metode *deep learning*, seperti CNN. Pemodelan generatif merupakan pembelajaran tanpa pengawasan (*unsupervised*) dalam *machine learning* yang secara otomatis menemukan dan mempelajari keteraturan atau pola dalam data input sedemikian rupa sehingga model dapat digunakan untuk menghasilkan atau mengeluarkan contoh baru yang secara masuk akal dapat diambil dari kumpulan data asli.

Conditional generative adversarial network atau singkatnya cGAN, adalah jenis GAN yang melibatkan generasi gambar bersyarat oleh model generator [1]. Dalam cGAN, pengaturan bersyarat diterapkan, artinya generator dan diskriminator dikondisikan pada semacam informasi tambahan (seperti label kelas atau data) dari modalitas lain. Akibatnya, model yang ideal dapat mempelajari pemetaan multi-modal dari input ke output dengan diberi informasi kontekstual yang berbeda.

Pix2pix merupakan salah satu model cGAN yang mempelajari pemetaan dari gambar input ke gambar output [2]. Pix2pix tidak terbatas pada satu aplikasi saja, model ini dapat diterapkan ke berbagai tugas, termasuk mensintesis foto dari peta label, menghasilkan foto berwarna dari gambar hitam putih, mengubah foto Google Maps menjadi gambar udara, dan bahkan mengubah sketsa menjadi foto.

II. Model dan Dataset

A. Pix2pix

Pix2pix atau *pixel to pixel* berarti dalam sebuah gambar dibutuhkan satu piksel, lalu mengubahnya menjadi piksel lain. Tujuan dari model ini adalah untuk mengkonversi dari satu gambar ke gambar lain, dengan kata lain tujuannya adalah untuk mempelajari pemetaan dari gambar input ke gambar output.

Arsitektur Pix2Pix GAN melibatkan spesifikasi model generator, model diskriminator, dan prosedur optimasi model yang cermat. Model generator dan diskriminator menggunakan blok lapisan Convolution-BatchNormalization-ReLU standar yang umum untuk *deep convolutional neural networks* [3].

Arsitektur model U-Net digunakan untuk generator, dan bukan model *encoder-decoder* umum. Arsitektur generator *encoder-decoder* melibatkan pengambilan gambar sebagai input dan *downsampling* pada beberapa lapisan hingga lapisan *bottleneck*, di mana representasi kemudian di-*upsample* lagi pada beberapa lapisan sebelum mengeluarkan gambar akhir dengan ukuran yang diinginkan [2].

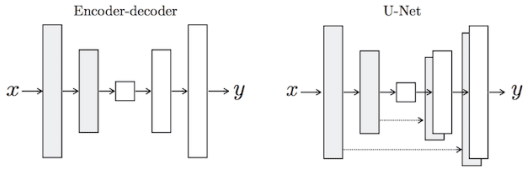

Fig. 1. Encoder-Decoder Generator dan U-Net Generator Models

Berbeda dengan model GAN tradisional yang menggunakan jaringan saraf convolutional dalam untuk mengklasifikasikan gambar, model Pix2Pix menggunakan PatchGAN. Model ini adalah *deep convolutional neural network* yang dirancang untuk mengklasifikasikan tambalan gambar input sebagai nyata atau palsu, bukan keseluruhan gambar. Model diskriminator PatchGAN diimplementasikan sebagai *deep convolutional neural network*, tetapi jumlah lapisan dikonfigurasi sedemikian rupa sehingga bidang reseptif yang efektif dari setiap output peta jaringan ke ukuran tertentu dalam gambar input. Output dari jaringan adalah peta fitur tunggal dari prediksi nyata/palsu yang dapat dirata-ratakan untuk memberikan skor tunggal [3].

B. Dataset

Basis data gambar yang digunakan dalam laporan ini adalah dataset *CMP Facade Database* [4]. Dataset ini menyajikan kumpulan data gambar fasad yang dikumpulkan di *Center for Machine Perception*, yang mencakup 606 gambar fasad yang diperbaiki dari berbagai sumber dan telah dianotasi secara manual. Fasad-fasad ini berasal dari berbagai kota di seluruh dunia dan gaya arsitektur yang beragam. Dataset terdiri dari 12 *class*.

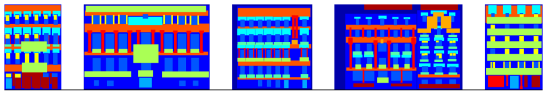

Fig. 2. Contoh gambar pada CMP Facade Database

III. Metodologi

A. Pre-processing

Sebelum melakukan *training* pada dataset, dilakukan *pre-processing* pada data. Seperti yang dijelaskan dalam paper

pix2pix, perlu diterapkan *jittering* dan *mirroring* acak untuk melakukan *pre-processing* pada set pelatihan. Untuk itu, didefinsikan beberapa fungsi, diantaranya adalah: 1) Mengubah ukuran setiap gambar 256×256 ke tinggi dan lebar yang lebih besar yaitu 286×286. 2) Meng-*crop* secara acak kembali ke 256×256. 3) Membalik gambar secara acak secara horizontal yaitu dari kiri ke kanan (*random mirroring*). 4) Normalisasikan gambar ke kisaran [-1, 1].

B. Networks

Setelah me-*load* dan melakukan *pre-processing* pada data, langkah pertama yang dilakukan adalah mendefinisikan beberapa fungsi bantuan untuk memuat set latih dan set uji. Setelah itu, dibangun sebuah generator. Generator untuk pix2pix cGAN adalah *modified U-Net*. U-Net terdiri dari encoder (downsampler) dan decoder (upsampler). Pada penelitian ini, random_normal_initializer untuk encoder dan decoder adalah 0. untuk *mean*, dan 0.02 untuk *stddev*. Ukuran kernel yang digunakan adalah 4 dengan *strides* = 2. Setelah downsampler dan upsampler didefinsikan, generator didefinsikan dengan hasil downsampler dan upsampler dengan menggunakan *OUTPUT_CHANNEL* = 4. Setelah itu, model arsitektur generator divisualisasikan. Terakhir adalah mendefinisikan *generator loss*. *Generator loss* adalah *sigmoid cross-entropy loss* dari gambar yang dihasilkan dan serangkaian gambar. Hal ini memungkinkan gambar yang dihasilkan menjadi mirip secara struktural dengan gambar target. Rumus untuk menghitung total loss generator adalah gan_loss + LAMBDA * l1_loss, dimana LAMBDA = 100.

Setelah membangun generator, berikutnya dibuat diskriminator. Diskriminator dalam pix2pix cGAN adalah pengklasifikasi PatchGAN convolutional — yang mencoba mengklasifikasikan apakah setiap *patch* gambar nyata atau tidak nyata. Diskriminator ini menerima dua input, yaitu gambar input dan gambar target, yang harus diklasifikasikan sebagai nyata, dan gambar input dan gambar yang dihasilkan (output dari generator), yang harus diklasifikasikan sebagai palsu. Setelah itu, model arsitektur diskriminator divisualisasikan. Langkah selanjutnya adalah mendefinisikan *discriminator loss*. Fungsi discriminator_loss mengambil 2 input yaitu gambar nyata dan gambar yang dihasilkan. real_loss adalah *sigmoid cross-entropy loss* dari gambar nyata dan array gambar sedangkan generate_loss adalah *sigmoid cross-entropy loss* dari gambar yang dihasilkan dari array nol. Total_loss adalah jumlah dari real_loss dan generate_loss.

Sebelum menghasilkan gambar, perlu didefinsikan *optimizer* dan *checkpoint-saver*. Ke-dua generator menggunakan fungsi optimasi *Adam* dengan beta_1 = 0.5. Gambar kemudian di-*plot* dengan membuat fungsi dimana fungsi tersebut menyampaikan gambar dari set uji ke generator, generator kemudian akan menerjemahkan gambar input menjadi output, dan langkah terakhir adalah mem-*plot* prediksi.

Langkah yang dilakukan dalam proses *training* adalah

- Untuk setiap *example input* menghasilkan *output*.
- Diskriminator menerima input_image dan gambar yang dihasilkan sebagai input pertama. Input kedua adalah input_image dan target_image.
- Selanjutnya, hitung *loss* dari generator dan diskriminator.
- Kemudian, hitung gradien kerugian sehubungan dengan generator dan variabel diskriminator (input) dan terapkan pada *optimizer*.
- Terakhir, lampirkan *loss* ke TensorBoard.

IV. Results and Conclusion

Dengan model Pix2Pix yang sudah dilatih, generator dapat menghasilkan gambar target baru yang diberikan dari gambar input. Fig. 3 menunjukkan beberapa contoh gambar yang dihasilkan.

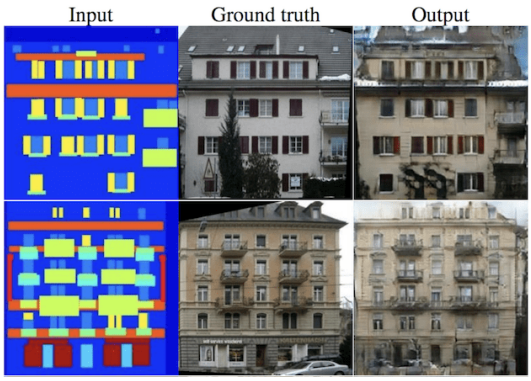

Fig. 3. Gambar yang dihasilkan menggunakan Pix2Pix

Penulis mengeksplorasi dan menganalisis pengaruh konfigurasi model yang berbeda dan *loss function* pada kualitas gambar.Temuan dari eksperimen ini menjelaskan mengapa pendekatan Pix2Pix efektif di berbagai *task* penerjemahan gambar.

Eksperimen dilakukan untuk membandingkan berbagai *loss function* yang digunakan untuk melatih model generator. Diantaranya adalah hanya menggunakan *L1 loss*, hanya *conditional adversarial loss*, hanya menggunakan *nconditional adversarial loss*, dan kombinasi L1 dengan setiap *adversarial loss.* Hasilnya menarik, menunjukkan bahwa L1 dan *conditional adversarial loss* saja dapat menghasilkan gambar yang masuk akal, meskipun gambar L1 buram dan gambar cGAN memperkenalkan artefak. Kombinasi keduanya memberikan hasil yang paling jelas.

Arsitektur model generator U-Net dibandingkan dengan arsitektur model generator *encoder-decoder* yang lebih umum. Kedua pendekatan dibandingkan dengan *L1 loss* saja dan *L1 + conditional adversarial loss,* menunjukkan bahwa *encoder-decoder* mampu menghasilkan gambar dalam kedua kasus, tetapi gambar jauh lebih tajam saat menggunakan arsitektur U-Net.

Eksperimen juga dilakukan untuk membandingkan diskriminator PatchGAN dengan bidang reseptif efektif yang berukuran beda. Versi model diuji dari bidang reseptif 1×1 atau PixelGAN, hingga 286×286 atau ImageGAN berukuran penuh, serta PatchGAN 16×16 dan 70×70 yang lebih kecil. Semakin besar bidang reseptif, maka semakin dalam jaringan. Ini berarti bahwa 1×1 PixelGAN adalah model yang paling dangkal dan

286×286 ImageGAN adalah model yang paling dalam. Hasil menunjukkan bahwa bidang reseptif yang sangat kecil dapat menghasilkan gambar yang efektif, meskipun ImageGAN berukuran penuh memberikan hasil yang lebih tajam, tetapi lebih sulit untuk dilatih. Menggunakan bidang reseptif 70×70 yang lebih kecil memberikan pertukaran kinerja (kedalaman model) dan kualitas gambar yang baik.

REFERENCES

[1] Abbasi, N., "What is a Conditional GAN (cGAN)?," *educative.io*, [Online]. Available: https://www.educative.io/answers/what-is-a-conditional-gan-cgan. [Accessed: June. 20, 2022].

[2] Isola, Phillip and Zhu, Jun-Yan and Zhou, Tinghui and Efros, Alexei A., "Image-to-Image Translation with Conditional Adversarial Networks." *arXiv*,2016, https://arxiv.org/abs/1611.07004.[Accessed: June. 20, 2022].

[3] Brownlee, J., "A Gentle Introduction to Pix2Pix Generative Adversarial Network," *machinelearningmastery*, July. 29, 2019. [Online]. Available: https://machinelearningmastery.com/a-gentle-introduction-to-pix2pix-generative-adversarial-network. [Accessed: June. 20, 2022].

[4] Radim Tyleček and Radim Šára, "Spatial Pattern Templates for Recognition of Objects with Regular Structure," Available: https://cmp.felk.cvut.cz/~tylecr1/facade/. [Accessed: June. 20, 2022].