

Penerapan Image Segmentation untuk Menghasilkan Efek Bokeh

Ahmad Farid Mudrika - 13522008¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

¹faridm0741@gmail.com

Abstrak— Efek bokeh merupakan salah satu efek paling umum digunakan oleh tidak hanya fotografer, tetapi semua pemotret foto. Efek ini dihasilkan dengan mengaburkan latar belakang suatu citra untuk menonjolkan objek utama dalam citra tersebut. Makalah ini membahas penerapan teknik *image segmentation* untuk memisahkan objek utama (*foreground*) dan latar belakang (*background*) dengan model DeepLabV3. Latar belakang lalu diterapkan operasi Gaussian blur untuk menghasilkan efek blur.

Kata Kunci — Citra, Image Segmentation, Deeplab, Bokeh.

I. PENGENALAN

Dalam dunia fotografi, salah satu poin penting untuk foto yang berkualitas tinggi adalah kemampuan fotografer untuk menonjolkan subjek utama dari objek-objek lain di sekitarnya. Salah satu teknik visual yang paling populer dan paling sederhana untuk mencapai tujuan ini adalah efek *bokeh*. Istilah *bokeh* berasal dari bahasa Jepang “boke” yang berarti kabur. Ini merujuk pada latar belakang yang tidak terfokus (out-of-focus) dalam sebuah foto. Efek ini memisahkan objek utama (*foreground*) dan latar belakang (*background*) dalam sebuah foto, memberikan kesan *depth*, dan mengarahkan fokus pelihat langsung ke objek utama.

Secara tradisional, efek bokeh dihasilkan melalui optik fisik menggunakan kamera dengan sensor berukuran besar dan lensa dengan bukaan (*aperture*) lebar. Namun, dengan masuknya perangkat seluler (*smartphone*) sebagai media fotografi paling populer, penciptaan efek *bokeh* menjadi tantangan. Hal ini karena sensor kamera *smartphone* yang relatif kecil jika dibandingkan dengan kamera tradisional. Akibatnya, pemisahan subjek foto dan latar belakang menjadi sulit dicapai tanpa lensa telephoto khusus.

Untuk mengatasi keterbatasan perangkat keras tersebut, pendekatan *Computational Photography* menjadi solusi yang semakin relevan. Pendekatan ini memanipulasi citra digital menggunakan algoritma perangkat lunak untuk mensimulasikan efek optik. Tantangan utama dalam menghasilkan bokeh sintesis yang realistis adalah akurasi dalam memisahkan area objek utama dan latar belakang. Metode pemisahan manual (seperti *masking* di Photoshop) memakan waktu dan tidak efisien untuk jumlah citra yang banyak. Oleh karena itu, diperlukan metode otomatis yang cerdas dan presisi.

Perkembangan teknologi *Computer Vision*, khususnya dalam bidang *Deep Learning*, telah menghadirkan solusi ampuh melalui teknik *Image Segmentation*. Berbeda dengan deteksi objek biasa yang hanya memberikan kotak pembatas (*bounding box*), segmentasi citra mampu mengklasifikasikan citra pada tingkat piksel (*pixel-level classification*). Salah satu

arsitektur jaringan saraf tiruan (*Neural Network*) yang menonjol dalam tugas ini adalah **DeepLabV3**. Model ini memanfaatkan *atrous convolution* untuk menangkap informasi konteks multi-skala, memungkinkan deteksi batas objek yang lebih halus dan akurat dibandingkan pendahulunya.

Makalah ini bertujuan untuk menerapkan teknik *Semantic Image Segmentation* menggunakan model DeepLabV3 untuk memisahkan *foreground* dan *background* pada citra tunggal (*monocular image*). Hasil segmentasi tersebut kemudian akan digunakan sebagai masker (*mask*) untuk menerapkan operasi pengaburan (*blurring*) menggunakan algoritma *Gaussian Blur* pada area latar belakang saja. Dengan metode ini, diharapkan dapat dihasilkan efek bokeh yang estetik dan menyerupai hasil kamera profesional secara otomatis, tanpa memerlukan perangkat keras optik yang memadai.

II. DASAR TEORI

A. Segmentasi Citra Semantik

Segmentasi citra adalah proses membagi citra menjadi beberapa segmen yang menyatakan objek citra. Segmentasi bertujuan untuk menyederhanakan representasi gambar agar lebih mudah dianalisis. Berbeda dengan *instance segmentation* yang membedakan setiap objek individu (orang A dan orang B dibedakan), *semantic segmentation* mengklasifikasikan tiap piksel dalam citra ke dalam kelas tertentu. Output dari proses ini adalah sebuah *mask* multikelas yang memetakan lokasi tiap objek yang dikenali.

B. DeepLabV3

DeepLabV3 adalah model jaringan saraf konvolusi (*Convolutional Neural Network / CNN*) yang dikembangkan oleh Google untuk tugas segmentasi semantik. Model ini dirancang untuk mengatasi tantangan dalam menangani objek pada berbagai skala (multiskala) dan mempertahankan resolusi spasial fitur.

DeepLabV3 menghasilkan output berupa *mask* segmentasi semantik, dengan tiap piksel diklasifikasikan ke dalam kelas tertentu sesuai dataset yang digunakan saat pelatihan. Umumnya, kelas yang dihasilkan meliputi 1 kelas latar belakang dan beberapa kelas objek, seperti manusia, hewan, kendaraan, dan objek-objek lainnya.

C. Gaussian Blur

Untuk mensimulasikan efek bokeh pada area latar belakang yang telah di-segmentasi, kami akan menggunakan operasi konvolusi dengan kernel Gaussian. Gaussian Blur adalah filter *low-pass* yang meratakan transisi intensitas piksel, menghasilkan efek buram yang halus.

Fungsi Gaussian dalam dua dimensi didefinisikan sebagai:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Dengan:

- x dan y adalah jarak dari titik asal sumbu horizontal dan vertikal.
- σ (sigma) adalah standar deviasi distribusi Gaussian, yang menentukan tingkat blur. Semakin besar nilai σ , semakin kuat efek bokeh yang dihasilkan

III. METODOLOGI

A. Lingkungan Implementasi

Implementasi sistem dilakukan menggunakan bahasa pemrograman Python. Beberapa pustaka (library) utama yang digunakan meliputi:

1. PyTorch: Sebagai kerangka kerja (framework) utama untuk memuat dan menjalankan model DeepLabV3.
2. Torchvision: Menyediakan model DeepLabV3 yang telah dilatih sebelumnya (pre-trained) pada dataset COCO/Pascal VOC.
3. OpenCV (Open Source Computer Vision Library): Digunakan untuk operasi manipulasi citra dasar seperti Gaussian Blur, resizing, dan konversi ruang warna.
4. NumPy: Digunakan untuk operasi matriks dan manipulasi array piksel.
5. Matplotlib: Digunakan untuk visualisasi hasil antara dan hasil akhir.

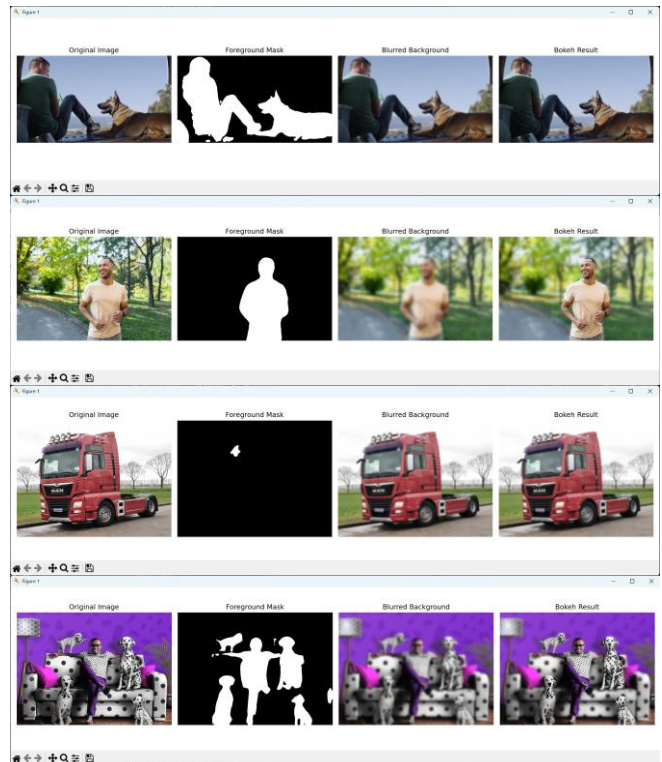
B. Implementasi

1. Pra-pemrosesan Citra (*Pre-processing*)
Sebelum citra dimasukkan ke dalam model DeepLabV3, citra harus disesuaikan dengan spesifikasi input jaringan saraf. Langkah ini meliputi:
 - Normalisasi: Mengubah nilai piksel citra (biasanya dalam rentang 0-255) menjadi rentang standar distribusi normal (mean dan standard deviation) yang sesuai dengan data latih model.
 - Resizing: Mengubah ukuran citra atau melakukan *padding* agar sesuai dengan dimensi tensor input model.
2. Segmentasi Menggunakan DeepLabV3
Citra yang telah diproses dimasukkan ke dalam model DeepLabV3. Model akan melakukan inferensi dan menghasilkan *output* berupa peta probabilitas (*probability map*) untuk setiap kelas yang dikenali. Untuk setiap piksel, model memprediksi kelas dengan probabilitas tertinggi.
3. Pembuatan Masker Biner (*Binary Mask Generation*)
Hasil keluaran model dikonversi menjadi masker biner M .

$$M(x, y) = \begin{cases} 1, & \text{jika piksel bukan latar belakang} \\ 0, & \text{jika piksel adalah latar belakang} \end{cases}$$
 Masker ini berfungsi sebagai "cetakan" untuk memisahkan area yang harus tetap tajam (bernilai 1) dan area yang akan dikaburkan (bernilai 0).

4. Penerapan Gaussian Blur Pada tahap ini, sistem membuat salinan dari citra asli dan menerapkan filter *Gaussian Blur* pada seluruh bagian citra salinan tersebut. Intensitas *blur* ditentukan oleh parameter kernel size dan standar deviasi. Hasilnya adalah citra yang sepenuhnya buram.
5. Rekonstruksi Citra (Blending) Langkah terakhir adalah menggabungkan kembali citra asli yang tajam dengan citra yang telah dikaburkan menggunakan masker sebagai panduan pencampuran (*blending*).

IV. PEMBAHASAN



Sistem telah diuji coba menggunakan beberapa sampel citra dengan karakteristik berbeda, mulai dari citra dengan objek manusia, hewan, dan kendaraan. Dari hasil percobaan, dapat dilihat bahwa model kesulitan mengenali truk sebagai subjek citra. Ini kemungkinan terjadi karena DeepLabV3 tidak dilatih untuk mengenali truk, melainkan hanya mobil biasa.

Pada kasus model dapat mengenali objek dengan baik, pada objek dengan tepi yang tegas seperti bahu, transisi area tajam dan blur terlihat rapi. Namun, pada kasus objek yang tertutup oleh objek lain, atau objek halus seperti bulu, terkadang terjadi kesalahan segmentasi. Kesalahan ini menyebabkan efek bokeh yang tidak tepat.

Penggunaan Gaussian blur memberikan efek buram yang merata. Meskipun teknik ini berhasil mengaburkan latar belakang dengan baik, efek yang dihasilkan terlihat berbeda dari efek bokeh yang dihasilkan lensa kamera asli. Namun, untuk tujuan estetika umum, terutama jika perangkat keras tidak mendukung efek bokeh secara *hardware*, efek bokeh yang dihasilkan tetap layak dipakai.

V. KESIMPULAN

A. Kesimpulan

Berdasarkan perancangan, implementasi, dan analisis pengujian yang telah dilakukan dalam makalah ini, dapat ditarik beberapa kesimpulan sebagai berikut.

1. Penerapan teknik Semantic Image Segmentation menggunakan arsitektur DeepLabV3 terbukti efektif untuk memisahkan objek utama (foreground) dan latar belakang (background) secara otomatis pada citra digital.
2. Penggabungan masker segmentasi dengan operasi Gaussian Blur mampu mensimulasikan efek bokeh (kedalaman bidang) yang secara visual estetik. Metode ini berhasil menonjolkan subjek utama dengan memburamkan latar belakang, meniru karakteristik fotografi lensa bukaan lebar (wide aperture) tanpa memerlukan perangkat keras optik khusus.
3. Meskipun secara umum hasil segmentasi cukup presisi pada bentuk tubuh manusia yang standar, sistem masih memiliki keterbatasan dalam menangani detail yang sangat halus (seperti helai rambut) atau objek dengan warna yang sangat mirip dengan latar belakang. Hal ini terkadang menimbulkan artefak visual di sekitar tepi objek.

B. Saran

Untuk pengembangan selanjutnya, penulis menyarankan hal berikut:

1. Penggunaan algoritma blur selain Gaussian Blur dengan algoritma yang lebih realistis secara optik.
2. Mengoptimalkan model agar sistem dapat dijalankan secara real-time pada *smartphone* atau webcam, bukan hanya pada citra statis

VI. UCAPAN TERIMA KASIH

Penulis ingin menyampaikan rasa terima kasih yang tulus kepada beberapa pihak atas kontribusi mereka terhadap makalah ini. Pertama-tama, penulis menyampaikan rasa syukur yang mendalam kepada Allah yang telah memberikan

petunjuk selama proses belajar dan penulisan. Selain itu, penulis juga mengakui dukungan dan pengajaran yang sangat berharga yang diterima dari dosen Strategi Algoritma IF4073 ITB, Bapak Rinaldi Munir. Pengetahuan dan bimbingannya telah memperkaya pengalaman belajar di kelas secara signifikan. Ucapan terima kasih yang istimewa juga ditujukan kepada keluarga dan teman-teman penulis atas dukungan mereka yang tak tergoyahkan sepanjang semester ini.

PRANALA GITHUB

Kode yang digunakan dan data yang lebih lengkap dapat dilihat di:

https://github.com/frdmmm/pcd_makalah2

DAFTAR PUSTAKA

- [1] <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2025-2026/22-Segmentasi-Citra-Bagian1-2025.pdf>
- [2] <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2025-2026/23-Segmentasi-Citra-Bagian2-2025.pdf>
- [3] <https://docs.pytorch.org/vision/main/models/deeplabv3.html>

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 24 Desember 2025



Ahmad Farid Mudrika 13522008