

# Pengaplikasian Algoritma *Divide and Conquer* Dalam Optimasi *R-CNN Semantic Segmentation*

Muhammad Rizky Sya'ban - 13521119<sup>1</sup>

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

<sup>1</sup>13521119@std.stei.itb.ac.id

**Abstrak**— Suatu citra dapat diolah menjadi sumber informasi jika diolah dengan Teknik tertentu. Salah satu bagian penting untuk memperoleh informasi yang lebih terperinci tentang objek atau struktur di dalam citra adalah dengan memisahkan piksel-piksel yang saling terkait menjadi kelompok-kelompok yang koheren dan bermakna. Salah satu Teknik pemisahan objek dari citra adalah dengan algoritma *Regional Convolution Neural Network (R-CNN)*, namun algoritma ini tidak efektif di semua kasus. Penggunaan algoritma yang tepat dapat mengoptimasi segmentasi gambar agar dapat lebih akurat. Salah satu algoritma yang dapat digunakan adalah Algoritma *Divide and Conquer (DnC)* yang membagi gambar menjadi beberapa wilayah lalu melakukan segmentasi terpisah pada masing-masing wilayah yang kemudian disatukan kembali menjadi segmentasi gambar yang utuh. Didapatkan bahwa algoritma DnC dapat mengoptimalkan proses segmentasi pada gambar-gambar dengan jumlah objek yang besar.

**Kata Kunci**—*Semantic Segmentation; OpenCV; Python; Divide and Conquer*

## I. PENDAHULUAN

Segmentasi citra, yang merupakan proses pemisahan piksel dalam citra menjadi kelompok atau segmen yang homogen berdasarkan atribut tertentu, memainkan peran yang sangat penting dalam perkembangan teknologi citra saat ini. Dalam bidang ilmu komputer, segmentasi citra memiliki peran yang signifikan dalam berbagai aplikasi seperti pengenalan pola, visi komputer, pengolahan medis, dan analisis citra.

Suatu citra dapat diolah menjadi sumber informasi jika diolah dengan Teknik tertentu. Salah satu bagian penting untuk memperoleh informasi yang lebih terperinci tentang objek atau struktur di dalam citra adalah dengan memisahkan piksel-piksel yang saling terkait menjadi kelompok-kelompok yang koheren dan bermakna.

Dalam rangka mencapai tujuan ini, algoritma pemisahan objek dari citra seperti *Regional Convolution Neural Network (R-CNN)* telah dikembangkan. Namun, terdapat situasi di mana algoritma ini tidak memberikan hasil yang efektif.

Untuk mengoptimalkan segmentasi gambar agar lebih akurat, diperlukan penggunaan algoritma yang tepat. Salah satu algoritma yang dapat digunakan untuk tujuan ini adalah Algoritma *Divide and Conquer (DnC)*. Algoritma ini mengadopsi pendekatan yang berbeda dengan membagi gambar menjadi beberapa wilayah yang lebih kecil. Setelah itu, dilakukan segmentasi terpisah pada setiap wilayah tersebut. Melalui pendekatan ini, proses segmentasi dapat dilakukan secara lebih fokus dan spesifik pada setiap bagian gambar.

## II. KAJIAN PUSTAKA

### A. Segmentasi Citra

Dalam pengolahan citra digital dan penglihatan komputer, segmentasi citra adalah proses pembagian citra digital ke dalam beberapa bagian (objek). Segmentasi citra bertujuan untuk menyederhanakan penggambaran citra ke dalam bentuk yang lebih bermakna dan lebih mudah dianalisis[1]. Metode segmentasi citra mencakup berbagai pendekatan, termasuk analisis intensitas, tekstur, dan warna citra. Pendekatan berbasis intensitas menggunakan perbedaan tingkat kecerahan piksel untuk memisahkan objek, sedangkan pendekatan berbasis tekstur memperhatikan pola dan struktur dalam citra. Pendekatan berbasis warna memanfaatkan perbedaan warna untuk mengklasifikasikan objek. Dengan hasil segmentasi citra yang akurat, informasi yang relevan dapat diekstraksi dan analisis lebih lanjut dapat dilakukan dengan lebih efisien dan akurat. Segmentasi Citra dapat dibagi menjadi beberapa tipe[6] :

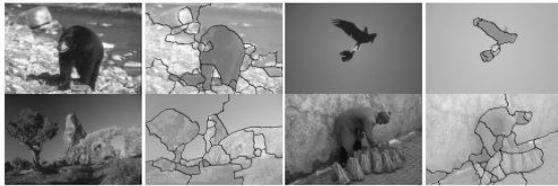
#### 1) *SuperPixels*

Metode ini membagi gambar berdasarkan kesamaan warna, tekstur, dan juga kecerahan, kemudian membentuk garis-garis kontur untuk masing-masing objek[4].

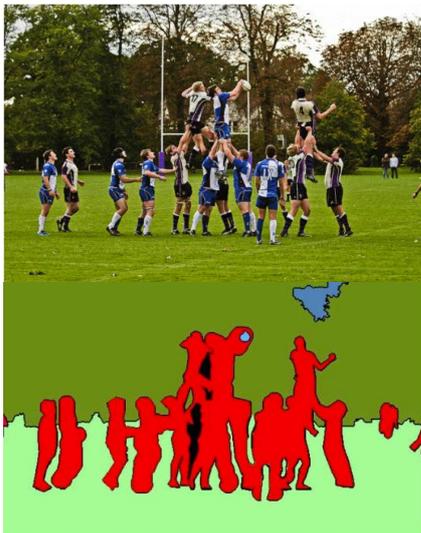
#### 2) *Semantic Segmentation*

Dalam Segmentasi Semantik, tujuannya adalah untuk menetapkan label (mobil, bangunan, orang, jalan, trotoar, langit, pohon, dll.) ke setiap piksel dalam gambar.

Kita dapat mengatakan piksel mana yang termasuk dalam kelas "orang" dengan hanya memeriksa apakah warna masker tersebut berwarna merah di piksel tersebut, tetapi kita tidak dapat mengatakan apakah dua piksel masker berwarna merah tersebut termasuk dalam orang yang sama atau orang yang berbeda.



Gambar 1 SuperPixels [4]



Gambar 2 Semantic Segmentation[6]

### 3) Instance Segmentation

konsep segmentasi ini erat kaitannya dengan Deteksi Objek. Namun, berbeda dengan Deteksi Objek, keluarannya adalah masker atau kontur yang berisi objek bukan sekedar kotak pembatas. Berbeda pula dengan Segmentasi Semantik, kami tidak melabeli setiap piksel dalam gambar; kami hanya tertarik untuk menemukan batas-batas objek tertentu.

### 4) Panoptic Segmentation

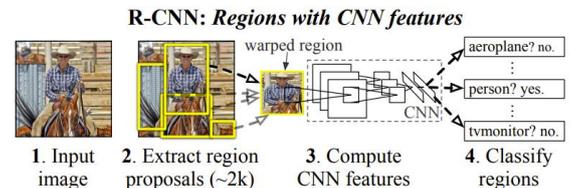
Segmentasi ini adalah gabungan dari *Semantic Segmentation* dan *Instance Segmentation*. Setiap piksel diberi label kelas, tetapi jika ada beberapa instansi dari kelas tersebut, kita tahu piksel mana yang termasuk ke instansi yang mana dari kelas tersebut.

## B. Regional Convolution Neural Network (R-CNN)

R-CNN adalah salah satu aplikasi awal yang sukses dan besar menggunakan jaringan saraf konvolusional untuk masalah lokalitas, deteksi, dan segmentasi objek. Pendekatan ini telah berhasil ditunjukkan pada dataset benchmark,

mencapai hasil terbaik pada saat itu pada dataset VOC-2012 dan dataset deteksi objek ILSVRC-2013 dengan 200 kelas[2].

R-CNN merupakan algoritma yang umum digunakan dalam mendeteksi objek-objek pada suatu citra/gambar. Dikarenakan suatu citra bisa terdapat beberapa objek baik yang sejenis ataupun tidak sehingga dibutuhkan suatu metode untuk membedakan objek-objek yang berada dalam suatu citra.



Gambar 3 R-CNN modul  
(Sumber: CVPR, 2014)

R-CNN dapat dibagi kedalam tiga modul[3], yaitu *Regional Proposal* yang dilakukan langsung pada gambar masukan dengan mendeteksi wilayah-wilayah yang berbeda dengan ukuran yang berbeda yang digambarkan dengan kotak-kotak pembatas. *Feature Extractor*, Setiap wilayah yang diusulkan akan dilatih oleh jaringan CNN, dan lapisan terakhir (4096 fitur) akan diekstraksi sebagai fitur sehingga output akhir dari Ekstraktor Fitur akan berupa Jumlah wilayah yang diusulkan x 4096. *Classifier*. Setelah fitur-fitur diekstraksi, kita perlu mengklasifikasikan objek di dalam setiap wilayah. Untuk melakukannya, model SVM linear dilatih untuk klasifikasi, khususnya satu model SVM untuk setiap kelas.

## C. Algoritma Divide and Conquer

Algoritma Divide and Conquer merupakan salah satu pendekatan dalam pemrograman yang memecah masalah yang kompleks menjadi beberapa submasalah yang lebih kecil. Kemudian beberapa submasalah tersebut diselesaikan secara terpisah lalu menggabungkan semua solusi submasalah menjadi solusi masalah asli. Sehingga secara umum algoritma dibagi menjadi tiga tahap[5] :

### 1) Divide

membagi persoalan menjadi beberapa upa-persoalan yang memiliki kemiripan dengan persoalan semula namun berukuran lebih kecil (idealnya berukuran hampir sama)

### 2) Conquer (Solve)

menyelesaikan masing-masing upa-persoalan (secara langsung jika sudah berukuran kecil atau secara rekursif jika masih berukuran besar)

### 3) Combine

mengabungkan solusi masing-masing upa-persoalan sehingga membentuk solusi persoalan semula.

### III. IMPLEMENTASI DAN PEMBAHASAN

#### A. R-CNN Semantic Segmentation

Proses pada algoritma *Semantic Segmentation* yang digunakan pada dasarnya terbagi menjadi empat tahap :

##### 1) Pemuatan Model

Pada makalah ini, penulis menggunakan Mask R-CNN model yang telah dikembangkan pada 2017 oleh Girshick, dkk[7]. Modul yang penulis gunakan adalah *frozen\_inference\_graph\_coco* beserta konfigurasinya.

##### 2) Mendeteksi objek berdasarkan kelasnya

Seluruh objek yang berhasil dideteksi oleh model tersebut diekstraksi kemudian diambil koordinat *mask*-nya agar bisa dilakukan tahap selanjutnya.

##### 3) Memberi warna pada masing-masing objek

Untuk setiap objek kemudian dibuat gambar baru sesuai kontur yang berada pada masing-masing *mask*-nya. Kemudian gambar baru yang menunjukkan batas-batas objek tersebut diberi warna sesuai kelasnya (*class\_id*)

##### 4) Penggabungan Objek

Dibuat suatu gambar polos dengan ukuran yang sama dengan gambar awal sebagai *background* yang diberi warna berbeda dari semua kelas. Kemudian gambar yang mewakili objek-objek bersesuaian digabungkan sesuai dengan koordinat awalnya.

#### B. Penerapan Algoritma Divide and Conquer

Untuk mengoptimalkan algoritma R-CNN, gambar masukan dapat dibagi menjadi beberapa bagian sebelum gambar disegmentasi. Idenya adalah algoritma ini memperkecil jumlah objek-objek yang harus dideteksi dalam satu kali proses segmentasi.

##### Pseudocode Algoritma Divide and Conquer

```
Function dnc_segmentation (image, threshold)->
image
{ Base case }
if image.shape below threshold
-> image

{ Rekurens }
else
{ #1 Divide}
subregions <- divide_image(image)

segmented_regions = []
for subreg in subregions
{ #2 Conquer(solve) }
segmented_subregion <- segment(subreg)

if segmented_subreg above threshold:
segmented_subregion <-
dnc_segmentation(segmented_subreg, threshold)
```

```
append(segmented_subregion)

{ #3 Combine }
combined_image = combine(segmented_reg)
-> combined_image
```

Fungsi diatas merupakan fungsi rekursif yang memiliki basis berupa gambar yang berada dibawah *threshold* atau Batasan minimum yang telah ditentukan dan rekurens berupa gambar yang masih berada diatas Batasan minimum yang ditentukan.

Pada rekurens, gambar akan dibagi menjadi empat bagian sama besar atau empat kuadran. Jika subregion tersebut berada dibawah *threshold*, akan dilakukan *conquer* yaitu dengan menerapkan algoritma R-CNN pada masing-masing *subregion*. Setelah semua bagian di-*solve* dilakukan combine dengan menggabungkan kembali *subregion* (yang bersesuaian dengan gambar sebelum diolah) tersebut menjadi gambar asalnya hingga terbentuk gambar hasil segmentasi yang sesuai dengan gambar awalnya.

#### C. Pengujian

Pada makalah ini akan diuji dua kemungkinan kasus yaitu gambar yang ramai yaitu memiliki objek dengan jumlah yang cukup besar (>20) dan juga gambar yang memiliki objek yang relatif kecil (<15). Pengujian pertama dan kedua akan mewakili kasus gambar-gambar yang memiliki banyak objek, sedangkan pengujian ketiga dan keempat mewakili kasus dimana gambar-gambar yang memiliki sedikit objek. Setiap gambar akan diuji dengan dan tanpa algoritma *divide and conquer* lalu dibandingkan jumlah objek yang berhasil terdeteksi oleh program. semua pengujian pada makalah ini menggunakan *threshold* 50% ukuran gambar.

##### 1) Pengujian pertama

Digunakan gambar New York Time Square pada malam hari dengan komposisi yang cukup heterogen. Gambar memiliki resolusi 1920x1200.



**Gambar 4** pegujian 1 Time-Square NYC

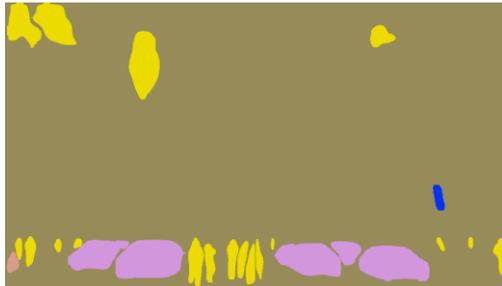
(Sumber: <https://wallpaperaccess.com/full/232806.jpg>)

Berikut adalah hasil pendeteksian objek oleh algoritma R-CNN. Kotak-kotak hijau menunjukkan

objek-objek yang berhasil dideteksi oleh algoritma R-CNN



**Gambar 5 Hasil pengujian 1: R-CNN tanpa DnC**  
Jumlah Objek : 23



**Gambar 6 Hasil Pengujian 2: R-CNN dengan DnC**  
Jumlah Objek : 33

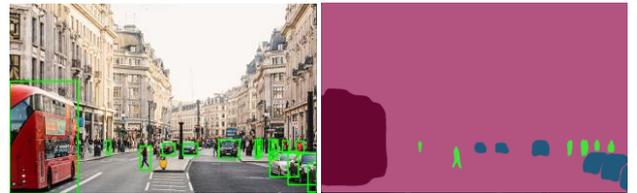
Dapat dilihat bahwa lebih banyak objek yang berhasil dideteksi oleh algoritma R-CNN ketika menggunakan *Divide and Conquer* dengan persentase 43.47 %

### 2) Pengujian 2

Digunakan gambar jalanan di kota London dengan mobil, bus, dan keramaian orang-orang. Gambar memiliki resolusi 1000×625



**Gambar 7 Pengujian 2 London Street**  
(Sumber: [telegraph.co.uk](http://telegraph.co.uk))



**Gambar 8 Hasil pengujian 2: R-CNN tanpa DnC**  
Jumlah Objek : 15



**Gambar 8 Hasil pengujian 2: R-CNN dengan DnC**  
Jumlah Objek : 31

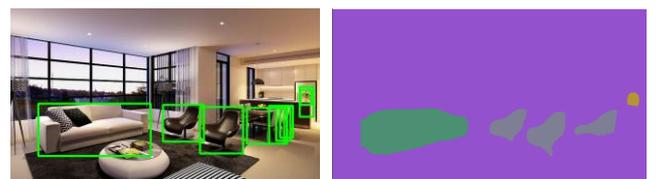
Dapat dilihat bahwa pada pengujian 2 penerapan Algoritma *Divide and Conquer* dapat dikatakan sangat optimal. dengan persentase 106.67 %

### 3) Pengujian 3

Digunakan gambar ruang tamu suatu rumah dengan objek-objek yang cenderung homogen. Gambar uji memiliki resolusi 728×410



**Gambar 9 Pengujian 3 Livingroom**  
(Sumber: [c4.wallpaperflare.com](http://c4.wallpaperflare.com))



**Gambar 8 Hasil pengujian 3: R-CNN tanpa DnC**  
Jumlah Objek : 15



**Gambar 9 Hasil Pengujian 3: R-CNN dengan DnC**  
**Jumlah Objek : 6**

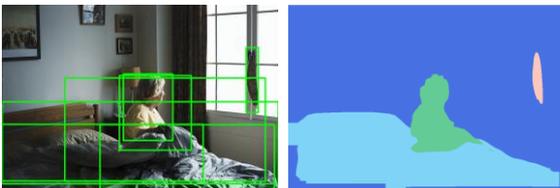
Dari hasil yang didapatkan pada pengujian 3, Algoritma Divide and Conquer tidak lagi optimal bahkan memperburuk akurasi segmentasi gambar dengan persentase -60%

#### 4) Pengujian 4

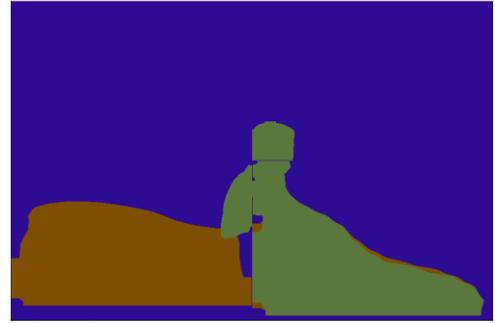
Pada pengujian keempat digunakan gambar yang lebih sederhana dibandingkan pengujian-pengujian sebelumnya agar didapatkan kasus dimana algoritma *Divide and Conquer* “memotong” objek dalam proses pembagian gambarnya. Gambar yang digunakan memiliki resolusi 910×607



**Gambar 10 Pengujian 4 Bedroom**  
 (Sumber: [bedroom image](#))



**Gambar 11 Hasil pengujian 3: R-CNN tanpa DnC**  
**Jumlah Objek : 7**



**Gambar 11 Hasil pengujian 3: R-CNN dengan DnC**  
**Jumlah Objek : 6**

Dari hasil yang didapatkan pada pengujian 4 ditemukan bahwa algoritma Divide and Conquer masih bisa mengenali objek yang sama walaupun objek terbagi dalam beberapa wilayah eksekusi. Walaupun begitu, terlihat bahwa dengan menggunakan *DnC* terdapat error dalam deteksi jumlah objek yang dihasilkan karena objek yang sebenarnya satu dianggap menjadi tiga pada kasus ini.

#### IV. KESIMPULAN DAN SARAN

Dari Keempat gambar yang telah diuji dapat dilihat bahwa algoritma *Divide and Conquer (DnC)* dapat mengoptimasi segmentasi gambar dengan *R-CNN dan Semantic Segmentation* pada beberapa kasus, yaitu ketika gambar yang ingin disegmentasi memiliki objek dengan jumlah yang besar atau objek-objek dengan skala yang kecil. Semakin banyak objek dalam satu gambar maka semakin optimal segmentasi menggunakan *R-CNN dengan Divide and Conquer*. Hal ini disebabkan karena terbatasnya kemampuan R-CNN dalam mendeteksi banyak objek dalam satu gambar sekaligus.

Ketika suatu gambar hanya memiliki beberapa objek saja dan dalam skala yang besar pada gambar, algoritma *Divide and Conquer* akan memberikan hasil yang tidak optimal dan bahkan memberikan error dalam pendeteksian objek. Sehingga semakin sederhana suatu gambar atau semakin sedikit objek yang berada dalam suatu gambar maka semakin buruk eksekusi dari algoritma *Divide and Conquer*. Salah satu penyebabnya adalah adanya *object truncating* yang menyebabkan objek yang sebenarnya satu dianggap lebih dari satu.

Tentunya algoritma yang penulis implementasikan masih dapat dikembangkan lagi sehingga menghasilkan solusi yang lebih optimal baik dengan modifikasi algoritma *Divide and Conquer* ataupun dengan algoritma yang lain.

LINK REPOSITORI GITHUB

[My github repository](#)

VIDEO LINK AT YOUTUBE

[Demo video](#)

UCAPAN TERIMA KASIH

Penulis ingin berterima kasih kepada bapak Dr. Ir. Rinaldi, M.T. dan seluruh tim pengajar IF2211 Strategi Algoritma yang telah memberikan penulis kesempatan dalam pembuatan makalah ini dan juga saya ingin berterima kasih kepada Girshick, dkk. yang telah menemukan algoritma *R-CNN*. Tak lupa, penulis mengucapkan terima kasih kepada orang tua, keluarga, dan teman-teman penulis atas dukungan dan do'anya.

#### REFERENSI

- [1] Linda G. Shapiro dan George C. Stockman. Computer Vision. Prentice-Hall, 2001.
- [2] R. Girshick, J. Donahue, T. Darrel, J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. CVPR, 2014.
- [3] R. Balasubramanian. Region – Based Convolutional Neural Network (RCNN) . Analytics Vidhya, 2021.
- [4] X. Ren, J. Malik. Learning a Classification Model for Segmentation. TTIC, 2003.
- [5] R. Munir. Algoritma Divide and Conquer Bagian 1. STEI ITB, 2021. (<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Divide-and-Conquer-%282021%29-Bagian1.pdf> diakses pada 22 Mei pukul 18.44 WIB)
- [6] S. Malik. Image Segmentation Using Computer Vision. LearnOpenCV, 2018.
- [7] K. He, G. Gkioxari, P. Dollar, R. Girshick. Mask R-CNN. Facebook AI Research, 2017.

#### 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, 22 Mei 2023



Muhammad Rizky Sya'ban  
13521119