

Laporan Tugas Besar Visi Komputer

Muhammad Ihsan Adly

Telkom University

Jl. Telekomunikasi No. 1, Terusan Buahbatu, Bojongsoang

muhammadihsanadly@student.telkomuniversity.ac.id

Abstract

Neural network telah memberikan pendekatan luar biasa terhadap hasil yang bisa diraih dalam pengerjaan tugas Visi Komputer. Namun hasil yang baik ini tidak mudah dicapai, karena hasil ini memakan biaya yang besar. Dalam paper ini diusulkan pendekatan Cross Stage Partial Network atau CSPNet untuk membantu meringankan kinerja komputasi. Dengan menggunakan CSPNet, kinerja komputasi turun hingga 20%

1. Introduction

Saat ini pendeteksian objek diharuskan menggunakan perangkat yang memiliki performa kinerja komputasi yang tinggi. Perangkat tersebut memiliki harga yang cukup mahal, sehingga orang-orang yang hanya memiliki perangkat biasa tidak mampu melakukan deteksi objek dengan baik. Dengan diusulkan nya CSPNet ini, diharapkan dapat mengurangi beban dari kinerja komputasi yang dibutuhkan untuk melakukan deteksi objek.

2. Related Work

Paper ini [1] penulis jadikan acuan karena proyek yang saya kerjakan tentang deteksi objek dibutuhkan kinerja komputasi yang tinggi, sedangkan perangkat yang digunakan penulis tidak dapat melakukan komputasi yang berat. Dengan menggunakan CSPNet, kinerja komputasi menjadi lebih ringan dan dapat menghasilkan performansi yang baik walaupun hanya dengan perangkat biasa.

3. Dataset

Dataset yang digunakan pada paper bersumber dari *ImageNet*. *ImageNet* sendiri mengandung dataset sejumlah 14,197,122 gambar yang beranotasi berdasarkan dari hirarki *WordNet*. Sebagai pembandingan, penulis menggunakan dataset yang berasal dari *Darknet*. *Darknet* sendiri terdiri dari 2 layer. Layer pertama berisikan data dari *Darknet* dan *Benign*, kemudian untuk layer dua

berisikan data dari VOIP, *Audio Stream*, *Browsing*, *Chat*, *File Transfer*, *Email*, P2P, dan *Video Stream*. Untuk pre-processing data tidak dilakukan sesuatu yang khusus.

4. Metode

4.1 Cross Stage Partial Network

Metode yang digunakan adalah *Cross Stage Partial Network* atau CSPNet.

DenseNet. pada setiap tahapan *DenseNet* mengandung sebuah *dense block* dan sebuah layer transisi, dan setiap *dense block* terdiri dari k *dense layers*

Cross Stage Partial DenseNet. sebuah tahapan CSPDenseNet terdiri dari sebagian *dense block* dan sebagian layer transisi. Dalam sebuah bagian *dense block*, *feature maps* dari layer dasar dalam sebuah tahapan dibagi menjadi dua bagian melalui chanel $x_0 = [x'_0, x''_0]$. Diantara x'_0 dan x''_0 , sebelumnya menjadi jalur langsung menuju tahapan akhir, dan kemudian akan melalui sebuah *dense block*.

Partial Dense Block. Tujuan merancang *partial dense block* adalah untuk : 1.)meningkatkan jalur gradien: Melalui strategi *split* dan *merge*, jumlah jalur gradien dapat digandakan. Karena strategi *cross-stage*, seseorang dapat meringankan kerugian yang disebabkan oleh penggunaan salinan peta fitur eksplisit untuk penggabungan; ; 2.)Perhitungan Keseimbangan setiap lapisan: biasanya, nomor saluran di lapisan dasar DenseNet jauh lebih besar daripada tingkat pertumbuhan. Semenjak saluran lapisan dasar yang terlibat dalam operasi *dense block* dalam akun *partial dense block* hanya untuk setengah dari aslinya jumlah, itu dapat secara efektif memecahkan hampir setengah dari *bottleneck* komputasi; 3.) mengurangi lalu lintas memori: Asumsikan ukuran peta fitur dasar dari *dense block* di DenseNet adalah $w \times h \times c$, tingkat pertumbuhannya adalah d , dan ada total m *dense layer*.

Partial Transition Layer. Tujuan merancang lapisan transisi parsial adalah untuk memaksimalkan

perbedaan gradien Kombinasi. Lapisan transisi parsial adalah mekanisme fusi fitur hierarkis, yang menggunakan strategi pemotongan aliran gradien untuk mencegah lapisan yang berbeda mempelajari informasi gradien duplikat.

Apply CSPNet to Other Architectures. CSPNet bisa dengan mudah diaplikasikan ke ResNet dan ResNeXt. Semenjak hanya setengah chanel fitur yang melalui Res(X)Blocks, maka tidak diperlukan menggunakan layer *bottleneck* lagi.

4.2 Exact Fusion Model

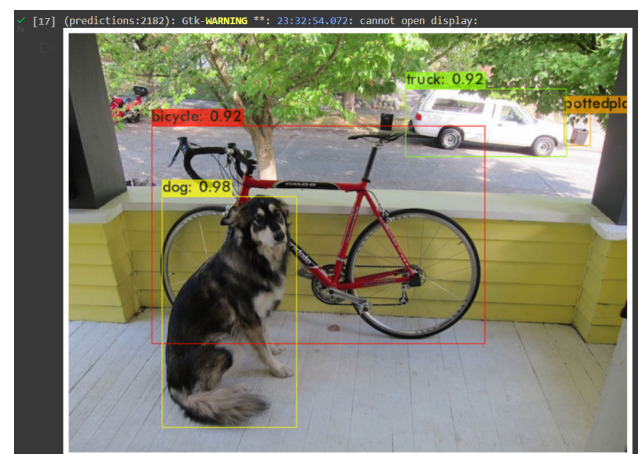
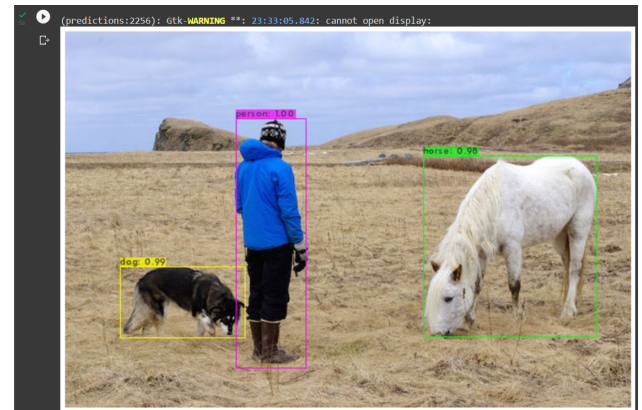
Looking Exactly to predict perfectly. Kami mengusulkan EFM yang menangkap Field of View (FoV) yang sesuai untuk masing-masing jangkak, yang meningkatkan akurasi detektor objek satu tahap. Untuk tugas segmentasi, karena label tingkat piksel biasanya tidak mengandung informasi global, biasanya lebih disukai untuk mempertimbangkan tambahan yang lebih besar untuk pengambilan informasi yang lebih baik. Namun, untuk tugas-tugas seperti klasifikasi gambar dan deteksi objek, beberapa informasi penting dapat menjadi tidak jelas ketika diamati dari label tingkat gambar dan tingkat kotak pembatas. CNN sering terganggu ketika belajar dari label tingkat gambar dan menyimpulkan bahwa itu adalah salah satu alasan utama bahwa detektor objek dua tahap mengungguli detektor objek satu tahap.

Aggregate Feature Pyramid. EFM yang diusulkan mampu menggabungkan piramida fitur awal dengan lebih baik. EFM nya berbasis dari YOLOv3, yang menetapkan tepat satu kotak pembatas sebelum setiap objek kebenaran dasar. Setiap tanah kotak pembatas kebenaran sesuai dengan satu kotak jangkak yang melampaui ambang batas IoU. Jika ukuran kotak jangkak adalah setara dengan FoV sel kisi, lalu untuk sel kisi skala S^{th} , kotak pembatas yang sesuai akan menjadi Dibatasi oleh skala $(S - 1)^{\text{th}}$ dan Upper dibatasi oleh skala $(S + 1)^{\text{th}}$. Oleh karena itu, EFM merakit fitur dari tiga skala.

Balance Computation. Karena peta fitur gabungan dari piramida fitur sangat besar, ini memperkenalkan sejumlah besar memori dan biaya komputasi. Untuk meringankan masalah, kami menggabungkan teknik Maxout untuk Kompres peta fitur.

5. Eksperimen

Hasil eksperimen yang penulis dapat dari menerapkan Algoritma CSPNet adalah sebagai berikut :



Akurasi *Object Detection* nya berada diatas 90%, namun fokus pengujian ini berada pada kecepatan *object detection* nya, melihat beberapa Assignment yang pernah penulis kerjakan, running program memakan waktu yang cukup lama, namun ketika menggunakan CSPNet hanya membutuhkan waktu beberapa detik saja, bahkan tidak sampai 15 detik.

6. Kesimpulan

Efisiensi yang dihasilkan saat menggunakan CSPNet meningkat drastis, waktu menjalankan program menjadi jauh lebih singkat walaupun menggunakan perangkat yang kinerja komputasi nya biasa saja. Dengan CSPNet beban kerja GPU atau CPU menjadi ringan, sehingga membuka kemungkinan untuk siapa saja melakukan penelitian terhadap *object detection*, karena tidak perlu memiliki atau menggunakan perangkat yang mahal. EFM yang diajukan menggabungkan *feature maps* yang dihasilkan dari *feature pyramid*, yang berkontribusi besar dalam mengurangi penggunaan memori. CSPNet dengan EFM yang diajukan sudah melampaui kinerja kompetitor dalam tingkat akurasi dan tingkat menyimpulkan hasil dalam *mobile GPU* atau CPU (komputer jinjing) dalam *object detection*.

References

- [1] Wang, C. Y., Liao, H. Y. M., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (pp. 390-391).