Speech Bubble Detection with Convolutional Neural Network, Canny Edge Detection and Run Length Smooth Algorithm

Ricky Setiawan Saswono, Rudy Adipranata, Kartika Gunadi
Program Studi Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra
Jl. Siwalankerto, 121-131 Surabaya 60236, Indonesia
Telp. (031) – 2983455, Fax. (031) - 8417658
rickysetiawan255@yahoo.com, rudya@petra.ac.id, kgunadi@petra.ac.id

ABSTRAK

Komik merupakan sebuah media hiburan yang biasanya digunakan untuk mengisi waktu kosong. Komik sendiri sudah sangat terkenal di dunia, terutama komik dari Jepang. Komik dari Jepang biasa disebut Manga, memiliki tingkat popularitas yang tinggi. Buktinya banyak sekali Manga yang diterjemahkan ke dalam Bahasa masing-masing tiap negara. Contohnya seperti One Piece yang sudah beredar di 43 negara. Walaupun begitu proses translasi cukup lama tertutama dalam penerjemahan Bahasa Jepang.

Penelitian ini dapat digunakan unutk mempercepat proses penerjemahan dengan cara menggunakan CNN dan Canny Edge Detection untuk mendeteksi balon ucapan pada Manga. Hasil deteksi tersebut disegmentasi dan dengan bantuan OCR untuk mendigitalisasi huruf Jepang. Kemudian menggunakan teknik *copy-paste* pada *online dictionary* atau *online translator* untuk mencari arti dari huruf yang tidak dimengerti. Karena mencari huruf dari kamus fisik (buku) memakan lebih banyak waktu.

Hasil penelitian untuk mensegmentasi balon ucapan dari Manga berhasil tetapi untuk menklasifikasikan gambar tersebut berupa balon ucapan atau bukan dengan CNN tidak berhasil. Peneliti berasumsi karena *dataset* yang dibuat jumlahnya sedikit atau masalah pada saat *pre-processing*.

Kata Kunci: CNN, Manga, RLSA, Canny Edge Detection, Balon ucapan.

ABSTRACT

Comic is an entertainment media that is usually used to fill free time. Comics themselves are already very well known in the world, especially comics from Japan. Comics from Japan, commonly called Manga, have a high level of popularity. The proof is a lot of Manga that is translated into each country's language. Examples such as One Piece that has been circulating in 43 countries. Even so the translation process is quite long especially in Japanese translation.

This research can be used to accelerate the translation process by using CNN and Canny Edge Detection to detect speech balloons in Manga. The detection results are segmented and with the help of OCR to digitize Japanese characters. Then use copy-paste techniques in an online dictionary or online translator to find the meaning of letters that are not understood. Because searching for letters from a physical dictionary (book) takes more time.

The results of the research to segment the speech balloon from Manga were successful but to classify the image in the form of a speech balloon or not with CNN was unsuccessful. Researchers assume because the dataset created is small in number or a problem during pre-processing.

Keywords: CNN, Manga, RLSA, Canny Edge Detection, Speech bubble.

1. PENDAHULUAN

Komik adalah media yang digunakan untuk mengekspresikan ide dengan gambar, sering dikombinasikan dengan teks atau informasi visual lainnya. Misalnya komik Jepang, yang disebut Manga dalam bahasa Jepang, memiliki balon ucapan yang berisi teks ucapan verbal karakter. Salah satu aplikasi pendeteksian elemen dalam gambar yang dipindai dari halaman komik adalah deteksi teks secara otomatis dan terjemahannya dari bahasa Jepang ke bahasa lokal [3]. Pada penelitian tersebut, metode Expert System dan Machine Learning yang diusulkan mendeteksi *sound effect* sebagai balon ucapan.

Pada penelitian yang lain [1][6], deteksi balon ucapan mendapatkan hasil yang bagus. Tetapi dengan *dataset* Manga109 yang merupakan kumpulan dari 109 Manga [5], hasil yang didapatkan tidak terlalu baik.

Penelitian ini berfokus untuk mendeteksi balon ucapan pada Manga dengan mensegmentasi kandidat balon ucapan dari halaman Manga dan diklasifikasikan menggunakan Convolutional Neural Network untuk menentukan kandidat balon ucapan merupakan balon ucapan atau bukan.

2. TINJAUAN PUSTAKA

2.1 Convolutional Neural Network

Convolutional Neural Network (CNN) adalah salah satu metode *machine learning* dari pengembangan Multi Layer Perceptron (MLP) yang didesain untuk mengolah data dua dimensi. CNN termasuk dalam jenis Deep Neural Network karena dalamnya tingkat jaringan dan banyak diimplementasikan dalam data citra. CNN memiliki dua metode; yakni klasifikasi menggunakan *feedforward* dan *backpropagation*. Cara kerja CNN memiliki kesamaan pada MLP, namun dalam CNN setiap *neuron* dipresentasikan dalam bentuk dua dimensi, tidak seperti MLP yang setiap *neuron* hanya berukuran satu dimensi.

2.2 Run Length Smooth Algorithm

Run Length Smooth Algorithm (RLSA) merupakan suatu metode yang berfungsi untuk mencari lokasi teks dalam suatu gambar citra biner. Cara kerja metode ini adalah dengan cara melakukan proses *scan-lines* pada gambar secara vertikal dan horizontal [4].

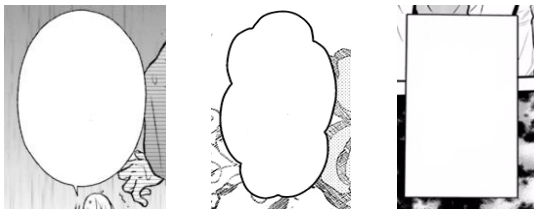
3. DESAIN SISTEM

3.1 Pembuatan Dataset

Karena *dataset* gambar balon ucapan yang dijadikan sebagai objek penelitian tidak ditemukan setelah dicari-cari melalui mesin pencari Google, maka peneliti membuat *dataset* tersebut secara pribadi. Proses pembuatan dataset berupa melakukan *crop* pada setiap balon ucapan, kemudian dari hasil *crop* tersebut dihilangkan teks yang berada di dalam balon ucapan dan latar belakang atau objek lain selain balon ucapan. Penghapusan latar belakang atau objek lain selain balon ucapan ini bertujuan untuk meningkatkan performa deteksi objek pada gambar [2]. Sumber balon ucapan tersebut diambil dari berbagai judul Manga yang disebarkan secara digital dan diakses secara gratis melalui internet.

Balon ucapan yang dipilih berupa balon ucapan berbentuk awan, oval, dan kotak. Balon ucapan yang dipilih tersebut merupakan balon ucapan yang paling umum digunakan. Contoh balon ucapan yang dipilih dan telah diproses dapat dilihat pada Gambar 1. Ada juga balon ucapan lain, tetapi balon ucapan tersebut kurang umum dan bentuknya dispesifikkan untuk suatu karakter atau kondisi tertentu.

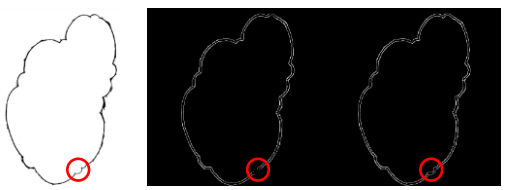
Dalam jangka waktu 1 bulan, bisa didapatkan sebanyak 750 gambar balon ucapan. Gambar-gambar tersebut terdiri dari 298 gambar berbentuk oval, 292 gambar berbentuk awan dan 160 gambar berbentuk kotak. Jumlah gambar yang didapatkan sedikit dikarenakan proses menghilangkan latar belakang dan objek lain selain balon ucapan membutuhkan waktu yang lama dan juga hanya dikerjakan sendiri.



Gambar 1. Contoh balon ucapan.

3.2 Pengolahan Dataset

Proses ini berupa mengolah gambar sebelum menjadi *input* untuk *training* CNN. Proses melibatkan algoritma *resize*, *sharpening* dan Canny Edge Detection untuk memproses gambar *grayscale*. *Resize* digunakan untuk mengubah ukuran resolusi gambar menjadi sama besar karena hasil *crop* berbeda-beda. Algortima *sharpening* digunakan karena saat gambar diolah hanya dengan algoritma Canny Edge Detection, terdapat beberapa gambar yang garis tepi pada balon ucapan hilang, tetapi pada gambar sebelum diproses tepi tersebut ada. Masalah ini dikarenakan pada algoritma Canny Edge Detection terdapat *Gaussian Blur* yang menyebabkan tepi tersebut dianggap lemah saat tahap menentukan potensi gradien gambar.



Gambar 2. Gambar dengan balon ucapan yang tepinya sebagian hilang.

Pada Gambar 2, dapat dilihat balon ucapan yang disebelah kiri, merupakan balon ucapan awal sebelum diproses dengan Canny Edge Detection. Tetapi pada balon ucapan yang berada ditengah, sebagian tepinya hilang setelah diproses hanya dengan menggunakan algoritma Canny Edge Detection. Berbeda dengan balon ucapan yang berada di sebelah kanan setelah di proses menggunakan algoritma Canny Edge Detection, tepinya tidak hilang. Hal ini dikarenakan sebelum diproses dengan algoritma Canny Edge Detection, gambar tersebut diproses terlebih dahulu dengan menggunakan algoritma *sharpening*. Dengan cara tersebut, tepi pada balon ucapan tersebut tidak hilang saat diproses dengan algoritma Canny Edge Detection. Kegunaan dari Canny Edge Detection sendiri untuk menghilangkan *noise* yang mungkin masih ada pada gambar *training* dan juga untuk memfokuskan garis tepi balon ucapan.

3.3 Segmentasi Kandidat Balon Ucapan

Untuk mendapatkan kandidat balon ucapan pada scan halaman Manga digunakan algoritma Run Length Smooth Algorithm (RLSA). Pada Manga yang umumnya semua halaman berwarna *grayscale*, tapi ada beberapa halaman yang biasanya diwarnai. Untuk mengatasi hal ini sebelum diproses lebih lanjut dilakukan *pre-processing* terlebih dahulu. *Pre-processing* yang dilakukan berupa mengubah warna RGB menjadi *grayscale*.

Tahap berikutnya, gambar halaman *scan* Manga akan diproses dengan RLSA untuk melakukan diskriminasi teks yang ada pada halaman *scan* Manga. Teks yang terdeteksi oleh RLSA akan menjadi blok hitam yang memanjang secara vertikal. Untuk mendapatkan posisi dari blok hitam hasil RLSA di gunakan *library* OpenCV yang memiliki fungsi findContours. *Contours* dapat dijelaskan sebagai kurva yang menghubungkan semua titik kontinu, memiliki warna yang sama atau intensitas yang sama. Hasil dari *Contours* merupakan kumpulan dari koordinat (x, y) yang akan membentuk Bounding Box. Gambar yang berada di area Bounding Box tersebut akan menjadi kandidat balon ucapan

3.4 Arsitektur Convolutional Neural Network

Arsitektur CNN yang digunakan adalah VGGNet, yang di kenalkan oleh Karen Simonyan dan Andrew Zisserman dari University of Oxford pada *paper* "Very Deep Convolutional Networks for Large-Scale Image Recognition". VGGNet sendiri merupakan *runner-up* dari kompetisi ILSVRC (ImageNet Large Scale Visual Recognition Competition) 2014.

Pada penelitian ini, peneliti menggunakan VGG-16 dikarenakan memiliki top-5 *error rate* paling kecil dibandingkan varian VGG yang lain, yaitu 8.8% [7]. VGG-16 terdiri dari 16 layer dan juga memiliki filter konvolusi berukuran 3x3 dibandingkan dengan filter konvolusi yang berukuran besar (5x5, 7x7), sehingga proses konvolusi menjadi lebih cepat. Untuk mengatasi *overfitting* yang biasa terjadi saat *training*, ditambahkan *dropout* setelah *layer* ke 10 dan 13.

3.5 Post-processing Potongan Gambar

Pada bagian ini, potongan gambar akan di dilakukan *sharpening* dan *bicubic interpolation*. *Sharpening* sendiri digunakan untuk menajamkan teks yang bewarna hitam dan *bicubic interpolation* untuk memperbesar ukuran gambar jika hasil potongan gambar memilki resolusi yang kecil. Sehingga saat di *zoom-in* gambar akan terlihat jelek dan susah untuk dilihat.

4. PENGUJIAN SISTEM

4.1 Pengujian Segmentasi Kandidat Balon Ucapan

Pada pengujian ini, gambar halaman Manga yang digunakan berasal dari Manga dengan judul "SPY x FAMILY". Halaman *sample* diambil sebanyak 5 halaman pertama dari *chapter* 1. Halaman dengan ilustrasi atau *cover* tidak termasuk karena tidak adanya balon ucapan pada halaman tersebut.

Tabel 1. Pengamatan balon ucapan pada halaman *sample*.

Halaman	Balon yang sesuai	Balon yang tidak sesuai	Total seluruh balon	Berwarna
1	6	0	6	Ya
5	3	2	5	Tidak
6	5	0	5	Tidak
7	2	4	6	Tidak
8	1	5	6	Tidak

Pada Tabel 1, dapat dilihat hasil pengamatan balon ucapan pada halaman Manga yang akan dijadikan *sample* pengujian.

Tabel 2. Hasil segmentasi dengan metode yang diusulkan.

Halaman	Jumlah Kandidat	Kandidat yang sesuai
1	17	2
5	37	5
6	43	8
7	42	4
8	35	1

Pada Tabel 2, hasil segmentasi kandidat balon ucapan dengan metode yang diusulkan berhasil mensegmentasikan balon ucapan pada halaman. Kandidat balon ucapan yang didapatkan tidak semuanya berupa balon ucapan, terdapat juga gambar lain seperti *background* ataupun bagian tubuh karakter. Pada kandidat balon ucapan dengan gambar balon ucapan yang didapatkan ada yang sempurna dan tidak sempurna. Tidak sempurna karena bagian balon ucapan terpotong sebagian atau Bounding Box terlalu besar. Walaupun begitu hasil metode segmentasi kandidat balon ucapan menurut peniliti sudah cukup baik untuk halaman yang berwarna *grayscale*. Untuk yang berwarna RGB metode yang diusulkan kurang baik.

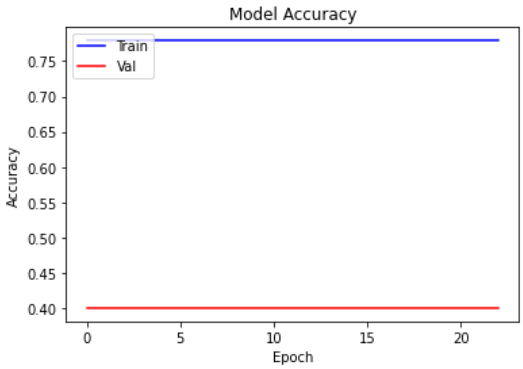
4.2 Pengujian dengan VGG-16

Pengujian dilakukan pada *platform* Google Colab dan menggunakan *runtime* GPU untuk proses *training* yang lebih cepat. Pada pengujian ini, proses *training* di tambahkan *checkpoint* dan *early stopping* yang berfungsi untuk menyimpan model yang paling bagus dan menghentikan proses *training* jika model yang di *training* tidak memberikan performa yang meningkat setelah beberapa *epoch*.

Dataset yang digunakan sudah diolah terlebih dahulu sebelum digunakan untuk *training* dan *test*. Resolusi gambar yang akan digunakan adalah 300x300 (dalam *pixel*). *Dataset* juga akan di acak dengan menggunakan *seed* 26416095, karena gambar pada *dataset* berurutan. Dari 750 gambar, 675 gambar digunakan untuk *training* dan 75 gambar untuk *test*. Jumlah gambar yang digunakan untuk *training* dan *test* berdasarkan nilai dari *validation_split* yang digunakan saat *training*, untuk pengujian ini nilai *validation split* sebesar 0.1.

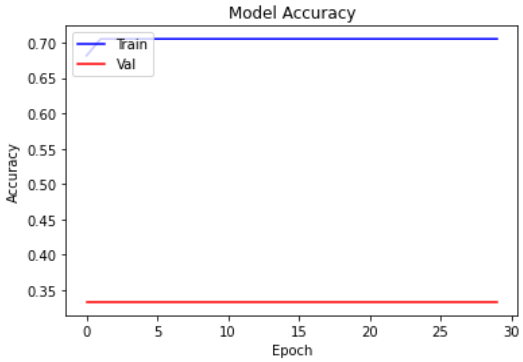
Pada pengujian ini nilai, *training epoch* sebanyak 30, *early stopping* yang digunakan adalah 20 dan *dropout* dengan nilai 0.2. Pada saat proses *training*, digunakan fitur *validation_split* yang berfungsi untuk melakukan *test* setelah proses *training* selesai setiap *epoch*.

Dapat dilihat pada Gambar 3, akurasi *test* tidak mengalami perubahan selama 22 *epoch*. Awalnya diasumsikan jika nilai *dropout* yang mempengaruhi proses *training*, sehingga *training* model yang tidak maksimal. Setelah dicoba tanpa menggunakn *dropout*, model masih memberikan akuras *test* yang sama pada model sebelumnya.

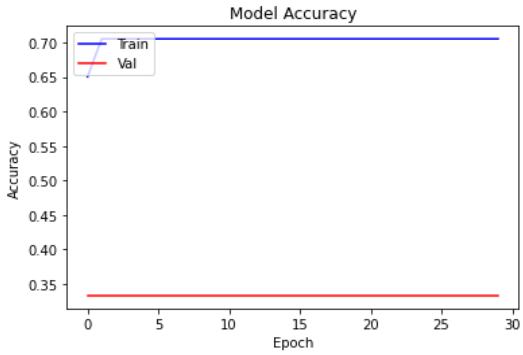


Gambar 3. Grafil akurasi model dengan *dropout* 0.2.

Peneliti mencoba untuk menggunakan *dataset* yang tidak diolah dan hanya melakukan *resize* saja. Dataset diacak dengan *seed* 26416095 dan dibagi menjadi 675 gambar untuk *training* dan 75 gambar untuk *test*. Training epoch sebanyak 30 dan tanpa menggunakan *early stopping*. Pengujian dilakukan dengan model yang menggunakan *dropout* 0.2 dan tanpa menggunakan *dropout*. Dapat dilihat pada Gambar 4 dan Gambar 5, akurasi *test* yang didapatkan oleh kedua model lebih rendah dari model yang menggunakan *dataset* yang telah diolah.

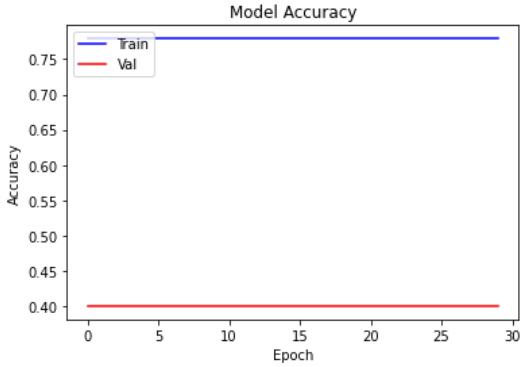


Gambar 4. Grafik model dengan nilai *dropout* 0.2.



Gambar 5. Grafik model tidak menggunakan *dropout*.

Peneliti mencoba untuk melakukan percobaan dengan menggunakan arsitektur yang berbeda, yaitu Inception V1 [8]. Dengan menggunakan *dataset* yang sama dan sudah diolah dan *training epoch* sebanyak 30 tanpa menggunakan *early stopping*. Pada Gambar 6, akurasi prediksi model tidak berbeda dengan model yang menggunakan arsitektur VGG-16.



Gambar 6. Grafik akurasi model Inception V1.

Peneliti mencoba untuk melakukan prediksi secara manual dengan model yang telah dibuat. Prediksi dilakukan terhadap *dataset training* dan *test*. Pada saat prediksi, didapatkan hasil yang menarik, yaitu semua kandidat gambar diprediksi sebagai gambar balon ucapan berbentuk awan.

4.3 Pengujian Klasifikasi

Model yang digunakan untuk melakukan prediksi adalah model yang menggunakan VGG-16 dengan *training epoch* 30 dan *dropout* 0.2. *dataset* yang digunakan untuk *training* dan *test* telah diolah dan diacak dengan *seed* 26416095.

Dapat dilihat pada Gambar 7, hasil prediki model selalu memberikan hasil yang sama, pada pengujian ini hasil tersebut berupa balon ucapan berbentuk awan. Tidak berbeda dengan model yang menggunakan arsitektur Inception V1, model juga memprediksi setiap gambar berupa balon ucapan berbentuk awan.

```
Gambar ke - 0.png prediksi 0 (
Gambar ke - 1.png prediksi 0 (
Gambar ke - 2.png prediksi 0 (
Gambar ke - 3.png prediksi 0 (
Gambar ke - 4.png prediksi 0 (
Gambar ke - 5.png prediksi 0 (
Gambar ke - 6.png prediksi 0 (
Gambar ke - 7.png prediksi 0 (
Gambar ke - 8.png prediksi 0 (
Gambar ke - 9.png prediksi 0 (
Gambar ke - 10.png prediksi 0
Gambar ke - 11.png prediksi 0
Gambar ke - 12.png prediksi 0
Gambar ke - 13.png prediksi 0
Gambar ke - 14.png prediksi 0
Gambar ke - 15.png prediksi 0
Gambar ke - 16.png prediksi 0
Gambar ke - 17.png prediksi 0
Gambar ke - 18.png prediksi 0
Gambar ke - 19.png prediksi 0
Gambar ke - 20.png prediksi 0
```

Gambar 7. Hasil prediksi VGG-16.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Segmentasi yang diusulkan untuk mendapatkan kandidat gambar balon ucapan dapat berjalan dengan baik dan memberikan hasil yang memuaskan, walaupun saat dilakukan pengamatan terhadap kandidat gambar balon ucapan yang berasal dari halaman Manga bewarna, banyak balon ucapan yang bermasalah. Tapi menurut peneliti, hasil yang diberikan sudah cukup baik terutama untuk gambar yang bewarna *grayscale*.

Sedangkan untuk CNN, mungkin arsitektur yang diusulkan dapat memberikan hasil yang baik juga. Karena arsitektur yang digunakan merupakan pemenang dan r*unner-up* dari kompetisi ILSVRC. Tetapi masih tidak diketahui penyebab utama model yang dibuat selalu memberikan hasil prediksi yang selalu sama. Peneliti berasumsi karena *dataset* yang dibuat kurang banyak atau ada masalah saat *pre-processin*g menyebabkan hasil prediksi CNN menjadi jelek.

5.2 Saran

Perlu dilakukan penelitian lebih lanjut untuk segmentasi pada gambar yang bewarna. Peneliti tidak melakukannya dikarenakan gambar berwarna pada Manga tidak dominan. Biasanya halaman berwarna hanya beberapa halaman saja untuk setiap volumenya dan sisanya bewarna *grayscale*. *Dataset* untuk *training* dapat ditambahkan lebih banyak dan dapat di *crop* ulang lagi karena hasil *crop* tidak *fit* dengan balon ucapan. P*re-processing* yang digunakan juga dapat diteliti ulang lagi, mungkin *pre-processing* yang digunakan kurang bagus.

6. DAFTAR PUSTAKA

[1] Dubray, D., & Laubrock, J. 2019. Deep CNN-based Speech Balloon Detection and Segmentation for Comic Books. *2019 International Conference on Document Analysis and Recognition (ICDAR)*. Sydney, Australia. 1237-1243. https://doi.org/10.1109/ICDAR.2019.00200

[2] Fang, W., Ding, Y., Zhang, F., & Sheng, V. 2019. DOG: A new background removal for object recognition from images. *Neurocomputing*, 361, 85-91. https://doi.org/10.1016/j.neucom.2019.05.095

[3] Kuboi, T. 2014. Element Detection in Japanese Comic Book Panels. Thesis, California Polytechnic State University, Computer Science, San Luis Obispo. doi:10.15368/theses.2014.141

[4] Liliana, Budhi, G. S., & Hendra. 2010. Segmentasi Plat Nomor Kendaraan Dengan Menggunakan Metode Run-Length Smearing Algorithm (RLSA). Retrieved from: https://www.researchgate.net/publication/277124943_Segmentasi_Plat_Nomor_Kendaraan_Dengan_Menggunakan_Metode_Run-Length_Smearing_Algorithm_RLSA

[5] Ogawa, T., Otsubo, A., Narita, R., Yusuke, M., Yamasaki, T., & Aizawa , K. 2018. Object Detection for Comics using Manga109 Annotations. arXiv:1803.08670v2. Retrieved from https://arxiv.org/abs/1803.08670

[6] Rigaud, C., Burie, J.-C., & Ogier, J.-M. 2017. Text-Independent Speech Balloon Segmentation for Comics and Manga. *International Workshop on Graphics Recognition*, 133-147. https://doi.org/10.1007/978-3-319-52159-6_10

[7] Simonyan, K., & Zisserman, A. 2015. Very Deep Convolutional Networks for Large-scale Image Recognition. arXiv:1409.1556v6**.** Retrieved from https://arxiv.org/abs/1409.1556

[8] C. Szegedy et al., Going deeper with convolutions, 2015 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, Boston, MA. 1-9, https://doi.org/10.1109/CVPR.2015.7298594