

PENGECAMAN KAPAL DAN BOT DARI IMEJ UDARA BERASASKAN RANGKAIAN SEMANTIK PENEMBERENGAN UNET

Looi Chiek Shen

Azizi Abdullah

Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia

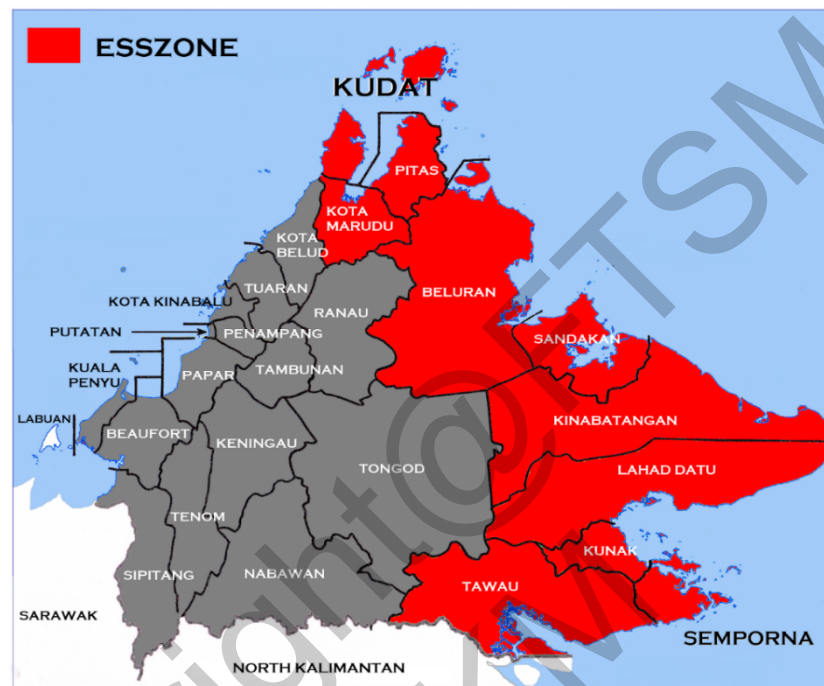
ABSTRAK

Selepas kejadian Lahad Datu pada tahun 2013, keselamatan di Eastern Sabah Security Zone (ESSZONE) merupakan sesuatu perkara yang membimbangkan penduduk tempatan dan kerajaan. Bagi mengurangkan risiko berlakunya kejadian ini berlaku, pembangunan algoritma penglihatan komputer yang boleh mengesan kapal dan bot telah dicadangkan. Masalah-masalah dengan algoritma yang sedia ada adalah imej-imej bot dan kapal daripada sudut pandangan satelit dan UAV adalah bersaiz kecil dan susah untuk dikecam. Objektif utama sistem ini adalah untuk membangun model untuk pengecaman kapal dan bot. Oleh itu, algoritma rangkaian konvolusion sepenuh telah dicadangkan untuk dilatih dan diuji untuk mengenali dan mengesan kapal dan bot. Set data untuk proses latihan adalah set data Seagull dan set data Airbus Ship Detection. Set data Seagull merupakan set data video yang telah dikutip menggunakan UAV manakala set data Airbus Ship Detection merupakan set data imej yang dikutip menggunakan satelit. Algoritma segmentasi objek yang digunakan adalah UNet. Kaedah pembelajaran mendalam akan diguna untuk menguji dan melatih set data yang telah dikumpulkan. Pembelajaran mendalam yang digunakan dalam algoritma ini adalah Vanilla Rangkaian Neural Konvolusional (CNN) yang telah diaplikasikan dan adalah berdasarkan struktur UNet. Tiga model telah dijalankan untuk mengkaji dan menguji ketepatan model. Ketiga-tiga model ini adalah model tanpa proses data augmentation, model dengan proses data augmentation dan model yang diubahsuai. Teknik data augmentation akan dicadang untuk meningkatkan prestasi dan ketepatan model. Selain itu, pengubaisuaian algoritma UNet dicadangkan dengan tujuan menguji perbezaan daripada segi ketepatan model. Dalam kajian ini, ketepatan model tanpa proses data augmentation telah mencapai ketepatan sebanyak 74.15%. Namun, ketepatan model bagi model dengan proses data augmentation dan model yang diubahsuai mempunyai ketepatan sebanyak 74.26% dan 73.14% masing-masing. Konklusinya, tiada perbezaan yang signifikan antara model dengan proses data augmentation, model tanpa proses data augmentation dan model yang diubahsuai.

1 PENGENALAN

Pada tahun 2013, pantai timur Sabah dan pulaunya telah diserang oleh pengganas Kesultanan Sulu. Pengganas-pengganas tersebut telah menyerang dan menculik penduduk-penduduk tempatan serta warga asing. Menurut statistik yang diberikan oleh komander ESSCOM pada tahun 2020 dan statistik yang diberikan oleh bekas Menteri di Jabatan Perdana Menteri, Datuk Seri Shahidan Kassim pada tahun 2016, terdapat 40 percubaan penculikan dari tahun 2018

hingga 2020 dan 15 percubaan penculikan dari tahun 2014 hingga 2016. Mereka merentas sempadan secara haram dengan menggunakan kapal dan bot. Beberapa minggu selepas pencerobohan itu, kerajaan dan angkatan tentera menubuhkan unit khas yang dikenali sebagai *Eastern Sabah Security Command* (ESSCOM). ESSCOM bertugas untuk mengawal Zon Keselamatan Sabah Timur (ESSZONE).



Rajah 1.1 Peta kawasan-kawasan ESSZONE

Kawasan-kawasan tersebut meliputi daerah Kudat, daerah Sandakan, daerah Lahad Datu, daerah Kunak dan daerah Tawau. Daerah-daerah ini mudah diserang oleh pegganas-pegganas. Oleh itu, pengesanan kapal dan bot adalah tugas penting di kawasan ESSZONE. Pengesanan kapal dan bot ini adalah penting untuk memastikan bahawa ESSCOM mempunyai pengawasan terhadap kapal-kapal yang berada di kawasan tersebut untuk menjaga keselamatan di kawasan-kawasan tersebut.

Bot dan kapal boleh dikesankan dengan bantuan pembelajaran mesin dan pembelajaran mendalam. Oleh itu, penggunaan pendekatan pembelajaran mesin dan pembelajaran mendalam. Pendekatan pembelajaran mendalam merupakan salah satu pendekatan yang sering diguna untuk mengesan objek. Pendekatan pembelajaran mendalam adalah pendekatan yang berdasarkan kepada rangkaian neural. Contoh rangkaian neural termasuklah Rangkaian Neural Konvolusional (CNN), Rangkaian Neural Berulang (RNN) dan *Long Short-Term*

Memory (LSTM). Walaupun ketiga-tiga contoh model ini adalah berdasarkan rangkaian neural tetapi ia mempunyai tugas-tugas yang berbeza. Dalam projek ini, CNN akan dipilih untuk melatih dan menguji set data imej. Model CNN dan algoritma segmentasi objek seperti Mask R-CNN, UNet dan SegNet akan dipilih untuk mengesan dan segmentasi objek-objek tertentu. Menurut penyelidikan Nie (2020), mereka menggunakan Mask R-CNN sebagai algoritma segmentasi kapal dan bot dan mereka juga melakukan perubahan terhadap Mask R-CNN. Model itu menggunakan FPN dan menambahkan *bottom-up path* untuk meningkatkan ketepatan Mask R-CNN. Model ini mencapai ketepatan sebanyak 76.1%. Selain itu, menurut kajian Karki dan Kulkarni (2021), UNet boleh digunakan sebagai algoritma segmentasi objek. Algoritma ini boleh digabungkan dengan *encoder* atau lapisan CNN yang lain seperti EfficientNet dan VGG.

2 PENYATAAN MASALAH

Dalam bidang pengesanan objek, terdapat beberapa cara dan teknik untuk mengesan objek. Namun, algoritma pengesanan objek mempunyai beberapa kelemahan. Antara kelemahan yang dikenal pasti adalah:

1. Set data yang sedia ada tidak sesuai untuk mengesan kapal dan bot menggunakan imej-imej UAV
 - Sudut rakaman UAV adalah sudut pandangan mata burung (*bird's eye view*) manakala model yang terlatih dengan set data *Imagenet* atau set data *COCO* adalah tidak sesuai untuk mengesan kapal dan bot. Kebanyakan kedua-dua set data ini menggunakan imej dan video daripada sudut pandangan mata manusia (*human's eye view*).
2. Kekurangan model yang terlatih dengan set data rakaman UAV
 - Kebanyakan model yang sedia ada sering dilatih dengan set data satelit sahaja. Contohnya, model-model yang sedia ada adalah terlatih dengan set data *Sentinel* atau dilatih dengan set data *Airbus*. Kedua-dua set data ini adalah set data satelit. Set data ini adalah tidak sesuai untuk digunakan untuk melatih model yang menggunakan UAV. Hal ini disebabkan set data satelit tidak diganggu oleh faktor-faktor pencahayaan.

3. Imej yang dirakamkan oleh UAV adalah kecil
 - Kaedah *sub-sample* yang digunakan dalam senibina CNN adalah kurang berkesan untuk mengesan objek bersaiz kecil

3 OBJEKTIF KAJIAN

Objektif utama dalam projek ini adalah:

1. Mengkaji teknik-teknik rangkaian konvolusi sepenuhnya untuk mengesan objek bersaiz kecil daripada imej UAV dan satelit.
2. Membangun model untuk pengesanan objek menggunakan imej UAV dan satelit.
3. Menguji ketepatan algoritma UNet.

4 METOD KAJIAN

4.1 Fasa Analisis

Fasa ini bertujuan untuk mengenal pasti tentang objektif projek, skop projek dan ciri-ciri projek. Fasa ini juga akan menumpukan perhatian terhadap kekangan dan masalah yang akan dihadapi. Dalam fasa ini, algoritma segmentasi objek yang akan digunakan telah dikenalpasti. Algoritma UNet akan diguna melatih dan menguji ketepatan model. Bahasa pengaturcaraan yang akan diguna adalah *Python*. Perisian yang perlu dipasang dalam komputer adalah Python, Pytorch, NVIDIA CUDA, NVIDIA CuDNN dan Anaconda.

4.2 Fasa Reka Bentuk

Fasa ini bertujuan untuk mereka bentuk algoritma dan sistem untuk projek ini. Spesifikasi, alat peralatan, set data dan bahasa pengaturcaraan yang digunakan dalam projek telah dikaji dan dirancang secara mendalam. Keperluan projek ini telah dikaji dan ditentukan. Berikut adalah keperluan spesifikasi mesin yang dicadang.

Jadual 4.1: Spesifikasi mesin yang paling minima untuk menjalankan model ini

Perkara	Spesifikasi paling minima
Sistem pengoperasian	Linux Distro
CPU	CPU yang mempunyai 2.40GHz dan ke atas
GPU	Kad grafik NVIDIA yang mempunyai VRAM 4GB dan diaktifkan dengan CUDA <i>compute capability</i> 3.0 dan ke atas

RAM	4GB dan ke atas
Kapasiti Simpinan	Sekurang-kurangnya 250GB

Penggunaan mesin yang mempunyai spesifikasi mengikut Jadual 4.1 adalah digalakkan. Mesin yang mempunyai spesifikasi tidak mengikut keperluan di Jadual 4.1 berpotensi untuk merosakkan mesin, terutamanya GPU. Hal ini disebabkan sistem yang akan dibangunkan akan menggunakan CUDA dan CuDNN untuk memproses imej dan video yang dimasukkan. Penggunaan sistem ini dengan GPU yang tidak mempunyai sokongan CUDA akan mengeluarkan *error* kerana sistem ini telah diaktifkan dengan CUDA.

Projek ini akan dibangunkan menggunakan spesifikasi mesin di bawah

Jadual 4.2: Spesifikasi mesin yang digunakan untuk membangunkan model ini

Perkara	Spesifikasi
Sistem pengoperasian	Ubuntu 20.04 <i>Long Term Support</i> (LTS)
CPU	Intel Core i5-3470, 3.20 – 3.60GHz, 4 Cores dan 4 Threads
GPU	NVIDIA GTX 1060 6GB
RAM	8GB 1600MHz
Kapasiti Simpinan	250GB SSD dan 1TB HDD

Jadual 4.2 menunjukkan spesifikasi mesin model yang akan diguna untuk membangunkan model ini.

Sistem ini mempunyai beberapa keperluan, antaranya adalah:

- a. Pengguna perlu memuat naik imej ke sistem untuk diproses oleh sistem ini.
- b. Sistem ini akan memaparkan segmentasi kapal dan bot sahaja
- c. Sistem ini akan memproses imej yang diambil daripada sudut satelit dan UAV sahaja
- d. Sistem ini akan mengesan imej yang mengandungi kapal dan bot pada waktu siang sahaja
- e. Sistem ini akan menerima fail JPEG (.jpg) dan PNG (.png) sahaja
- f. Imej-imej *mask* adalah hasil daripada sistem ini
- g. Imej yang dimuat naik ke sistem haruslah berwarna RGB

Bagi menjalankan proses latihan dalam algoritma UNet, terdapat beberapa syarat dan keperluan yang perlu dipenuhi sebelum proses latihan dijalankan. Antaranya adalah:

- a. Proses latihan algoritma perlu mempunyai imej asal dan imej *masked*.
- b. Imej asal yang digunakan untuk proses latihan harus berwarna RGB sahaja.
- c. Imej *masked* yang digunakan perlulah dalam format *binary* sahaja.
- d. Saiz imej asal dan imej *masked* yang digunakan dalam proses latihan haruslah mempunyai saiz yang sama.

Selain itu, aliran kerja model dan GUI juga direkabentukkan.

4.3 Fasa Pembangunan

Dalam fasa ini, pembangunan model akan dijalankan dan diagihkan kepada beberapa fasa iaitu:

1. Fasa pengutipan data
2. Fasa pemprosesan data
3. Fasa pembinaan model
4. Fasa latihan model
5. Fasa pembangunan GUI

Proses-proses ini akan dijalankan mengikut fasa dan turutan yang telah ditetapkan.

Proses latihan dan pengujian akan menggunakan set data *Seagull* (Ribeiro et al., 2017) dan set data *Airbus Ship Detection* (2017). Selepas itu, proses pengujian akan dijalankan menggunakan set data yang sama. Set data ini mengandungi imej-imej dengan kapal dan imej-imej tanpa kapal. Setiap posisi kapal akan disegmentasikan dan *masked image* akan dihasilkan. Set data *Seagull* mengandungi imej yang diambil daripada UAV dan set data *Airbus Ship Detection Dataset* mengandungi imej yang diambil daripada satelit.

Set data *Seagull* telah diperolehi daripada *University of Lisbon*, Portugal dan *Portuguese Air Force Research Centre*. Kamera GoPro 2 yang telah dipasang ke UAV untuk merekod video-video. Kamera ini menghasilkan video pada resolusi 1920 x 1080 *pixels*. Penggunaan UAV membolehkan kamera untuk mengutip video dan imej dari sudut pandangan mata burung (*bird's eye point of view*). Selain itu, kamera LWIR juga digunakan dalam pengutipan video dan imej untuk set data. Namun, set data infrared tidak akan digunakan dalam pembangunan model dan pengujian. Set data *Seagull* mempunyai sebanyak 11,400 imej. Set data ini adalah set data yang tidak mempunyai anotasi. Oleh itu, set data ini perlu menjalani proses-proses anotasi imej.

Set data *Airbus Ship Detection* (2017) juga digunakan untuk tujuan latihan dan pengujian model. Set data ini telah digunakan untuk menangani masalah-masalah kekurangan data atau imej untuk tujuan latihan dan pengujian. Selain itu, set data ini juga telah digunakan untuk meningkat kepelbagaian data atau imej untuk tujuan latihan dan pengujian. Set data ini mengandungi sebanyak 193,000 imej. Namun, hanya 32,000 imej telah dipilih untuk tujuan pembangunan model dari segi latihan dan pengujian model. Hal ini disebabkan proses latihan dan pengujian untuk set data yang mempunyai 193,000 imej memerlukan sumber yang banyak, misalnya dari segi komputational. Set data ini adalah set data yang telah dianotasi. Oleh itu, set data ini tidak perlu dianotasi.

Jadual 4.3 memaparkan agihan set data atau imej yang digunakan mengikut set data dan kategori set data.

Jadual 4.3: Agihan set data yang digunakan

Jenis Set Data	<i>Seagull</i>	<i>Airbus Ship Detection</i>	Jumlah Imej
Imej dengan kapal atau bot	7,300	16,377	23,677
Imej tanpa kapal atau bot	4,100	20,492	24,592
Jumlah Imej	11,400	36,996	48,629

Dalam fasa pembangunan juga, set data telah dianotasi dan penjanaan imej *masked* telah dilaksanakan. Kedudukan objek dan jenis objek dalam imej dan video yang dicari juga akan dianotasi dan dilabel. Untuk menganotasi dan menglabel set data, perisian *VGG Image Annotator* akan digunakan untuk menjalani tugas-tugas tersebut. Koordinat segmentasi dan jenis objek dalam imej dan video akan dihasilkan dalam bentuk fail JSON (.json) dan CSV (.csv). Koordinat segmentasi yang dihasil akan digunakan untuk menghasil imej *maksed*.

Selain itu, set data telah diagih kepada *training* dan *testing*.

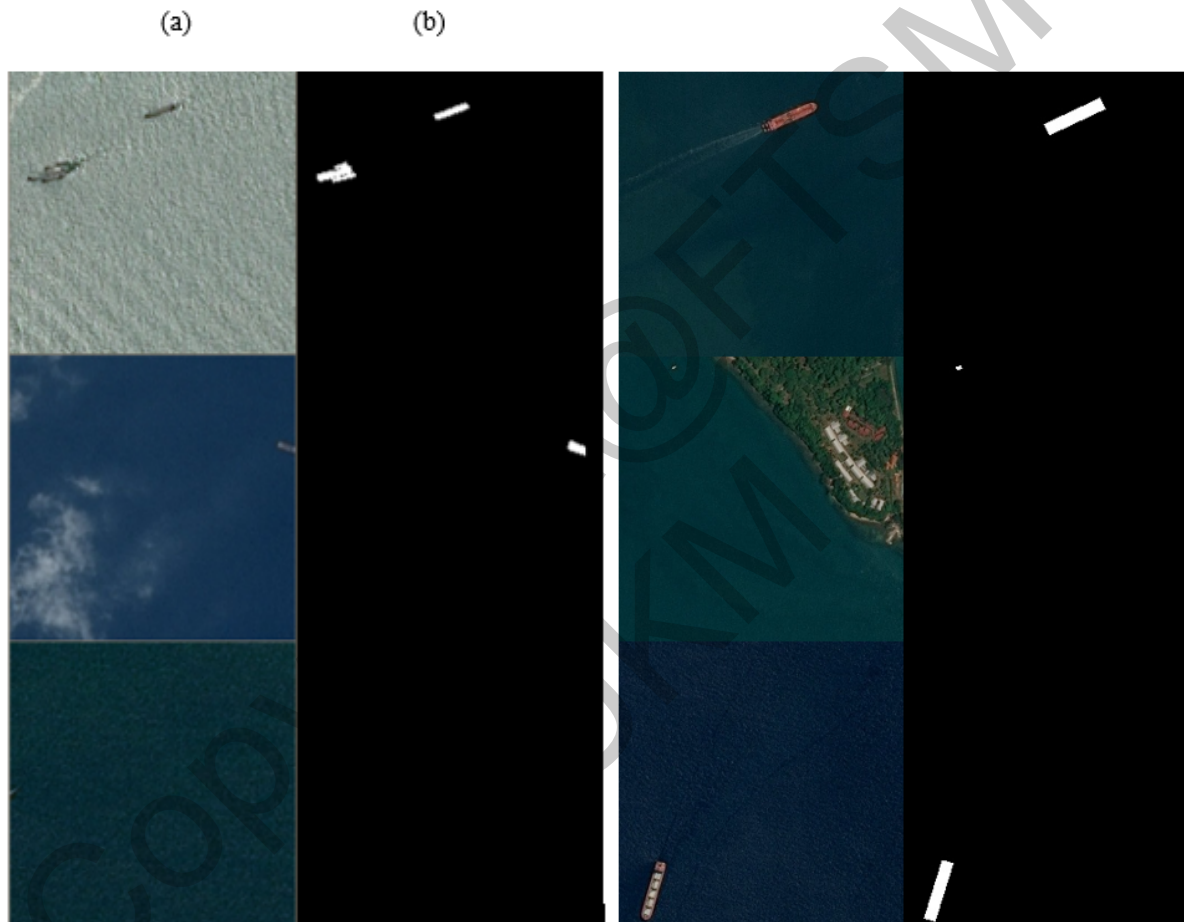
Jadual 4.4 menunjukkan pengagihan set data kepada set data latihan, pengujian dan pengujian.

Jadual 4.4: Bilangan set data dan peratusan mengikut pengagihan set data latihan dan pengujian

Jenis set data	Set data latihan	Set data pengujian

Peratusan set data	70%	30%
Bilangan imej dalam set data	33,788	14,481

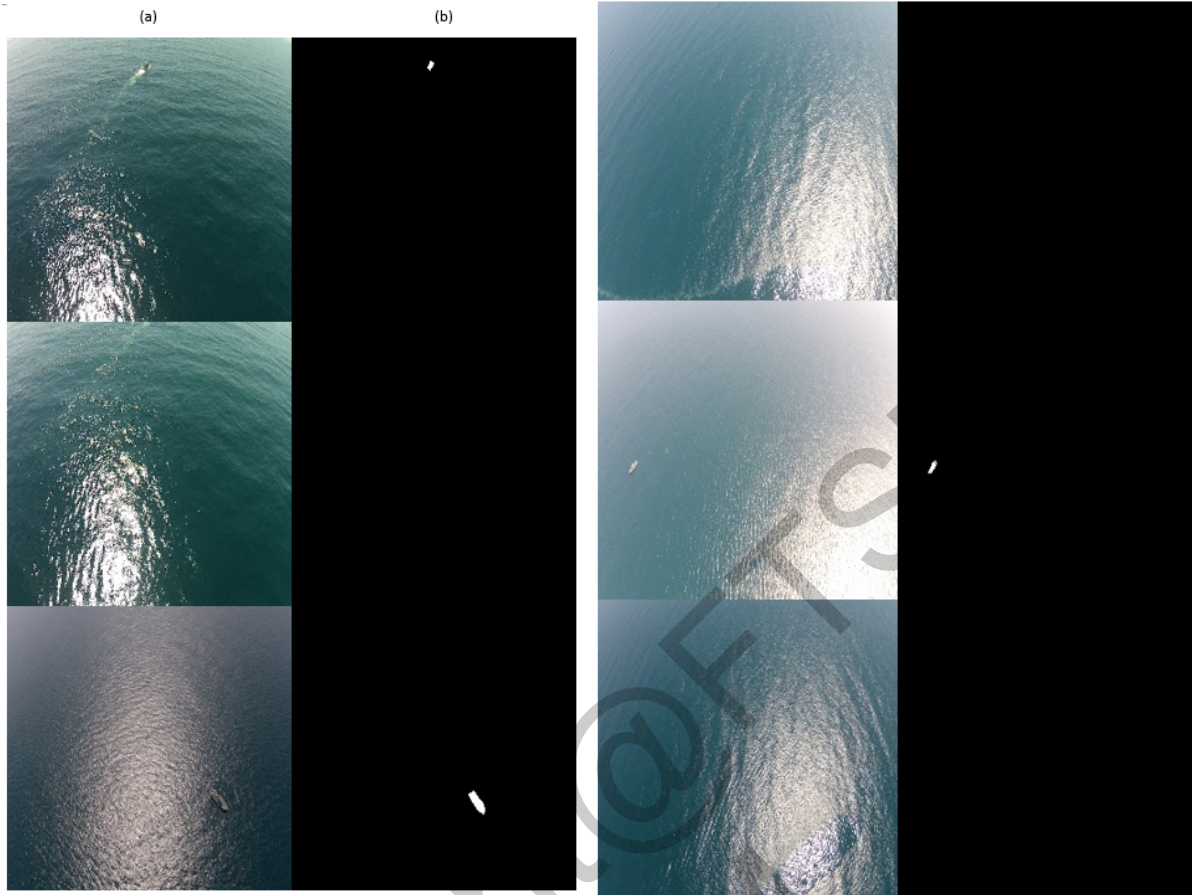
Rajah 4.1 dan Rajah 4.2 adalah contoh imej yang perlu digunakan pada proses latihan model. Rajah 4.1 menunjukkan imej sebenar dan imej *masked* dalam set data *Airbus Ship Detection*. Selain itu,



Rajah 4.1: Imej-imej sebenar dan imej-imej *masked* daripada set data *Airbus Ship Detection*

Rajah (a) dalam Rajah 4.1 menunjukkan imej asal dalam set data *Airbus Ship Detection*.

Rajah (b) dalam Rajah 4.1 menunjukkan imej *masked* yang dijanakan untuk proses latihan.



Rajah 4.2: Imej-imej sebenar dan imej-imej *masked* daripada set data *Seagull*

Rajah (a) dalam Rajah 4.2 menunjukkan imej sebenar dalam set data *Seagull* manakala rajah (b) dalam Rajah 4.2 menunjukkan imej *masked* yang dihasilkan selepas imej-imej telah dianotasi. Imej-imej *masked* akan digunakan proses latihan dalam set data *Seagull*.

Pengaturcaraan sistem juga telah dilaksanakan mengikut spesifikasi yang ditetapkan. Fasa ini telah melibatkan pengaturcaraan algoritma segmentasi semantik objek. Selepas membangunkan algoritma segmentasi semantik objek, set data yang telah diproses akan digunakan untuk melatih baharu supaya boleh mengesan bot dan kapal. Model UNet akan digunakan untuk melatih dan menguji set data. Jadual 4.5 menunjukkan senarai lapisan dalam struktur UNet.

Jadual 4.5: Senarai lapisan-lapisan dalam struktur UNet

No	Type	Input	Filter	Size and Stride	Output
1	conv1	256 x 256 x 2	64	3 x 3 / 1	256 x 256 x 64
2	conv2	256 x 256 x 64	64	3 x 3 / 1	256 x 256 x 64
3	max	256 x 256 x 64		2 x 2 / 2	128 x 128 x 64
4	conv	128 x 128 x 64	128	3 x 3 / 1	128 x 128 x 128
5	conv	128 x 128 x 128	128	3 x 3 / 1	128 x 128 x 128

6	max	128 x 128 x 128		2 x 2 / 2	64 x 64 x 128
7	conv	64 x 64 x 128	256	3 x 3 / 1	64 x 64 x 256
8	conv	64 x 64 x 256	256	3 x 3 / 1	64 x 64 x 256
9	max	64 x 64 x 256		2 x 2 / 2	32 x 32 x 256
10	conv	32 x 32 x 256	512	3 x 3 / 1	32 x 32 x 512
11	conv	32 x 32 x 512	512	3 x 3 / 1	32 x 32 x 512
12	max	32 x 32 x 512		2 x 2 / 2	16 x 16 x 512
13	conv	16 x 16 x 512	1024	3 x 3 / 1	16 x 16 x 1024
14	conv	16 x 16 x 1024	1024	3 x 3 / 1	16 x 16 x 1024
15	de-conv	16 x 16 x 1024	512	3 x 3 / 2	32 x 32 x 512
16	concat	32 x 32 x 512			32 x 32 x 1024
17	conv	32 x 32 x 1024	512	3 x 3 / 1	32 x 32 x 512
18	conv	32 x 32 x 512	512	3 x 3 / 1	32 x 32 x 512
19	de-conv	32 x 32 x 512	256	3 x 3 / 2	64 x 64 x 256
20	concat	64 x 64 x 256			64 x 64 x 512
21	conv	64 x 64 x 512	256	3 x 3 / 1	64 x 64 x 256
22	conv	64 x 64 x 256	256	3 x 3 / 1	64 x 64 x 256
23	de-conv	64 x 64 x 256	128	3 x 3 / 2	128 x 128 x 128
24	concat	128 x 128 x 128			128 x 128 x 256
25	conv	128 x 128 x 256	128	3 x 3 / 1	128 x 128 x 128
26	conv	128 x 128 x 128	128	3 x 3 / 1	128 x 128 x 128
27	de-conv	128 x 128 x 128	64	3 x 3 / 2	256 x 256 x 64
28	concat	256 x 256 x 64			256 x 256 x 128
29	conv	256 x 256 x 128	64	3 x 3 / 1	256 x 256 x 64
30	conv	256 x 256 x 64	64	3 x 3 / 1	256 x 256 x 64
31	conv	256 x 256 x 64	3	1 x 1 / 1	256 x 256 x 3

Conv merupakan *convolution*, de-conv merupakan *de-convolution*, max merupakan *max-pooling* dan concat merupakan *concatenate layer*.

Pengujian-pengujian ini dijalankan untuk memastikan spesifikasi dan keperluan yang telah dinyatakan dalam fasa-fasa yang selepas dicapai. Pembangunan GUI telah dilaksanakan dalam fasa ini.

4.3.1 Model Tanpa Proses Data Augmentation

Proses latihan dalam model ini akan dijalankan tanpa proses data augmentation. Namun, proses prapemprosesan imej akan dijalankan semasa proses latihan model dijalankan. Proses prapemprosesan yang dijalankan terhadap imej adalah perubahan kontras imej dan perubahan kecerahan imej. Proses-proses ini hanya akan melibatkan imej asal sahaja.

Sebelum imej-imej dimuatkan untuk proses latihan dengan menggunakan modul *Dataloader* yang ditawarkan dalam PyTorch, pengekodan sebuah modul untuk memadamkan imej-imej asal dan imej-imej *masked* yang mempunyai nama yang sama perlu dipadankan. Selain itu,

modul ini juga berfungsi untuk menjalani prapemrosesan imej dan *data augmentation* terhadap imej jika model tertentu memerlukannya. Bagi memuatkan imej ke modul *Dataloader*, modul yang dibina ini perlulah digunakan untuk memuatkan imej-imej asal dan imej-imej *masked*.

4.3.2 Model Dengan Proses Data Augmentation

Proses-proses semasa latihan adalah sama dengan model tanpa proses data augmentation. Namun, proses *data augmentation* ditambahkan dalam model ini untuk dilaksanakan. Proses *data augmentation* ini dijalankan untuk menambahkan kepelbagaian imej untuk dilatih oleh algoritma UNet yang digunakan. Jadual 4.6 menunjukkan proses data augmentation yang dijalankan semasa proses latihan.

Jadual 4.6: Penggunaan kaedah-kaedah dalam pakej Torchvision untuk proses data augmentation

Kaedah	Jenis	Penerangan
Transformasi Geometri	<i>Vertical Flipping</i>	<i>Flip</i> imej secara menegak. Kebarangkalian berlakunya fungsi ini adalah 0.5
	<i>Horizontal Flipping</i>	<i>Flip</i> imej secara melintang. Kebarangkalian berlakunya fungsi ini adalah 0.5
<i>Random Cropping</i>	<i>Crop</i> imej	Imej akan dipotong kepada imej yang bersaiz 128 x 128 <i>pixels</i> secara rawak. Kebarangkalian berlakunya fungsi ini adalah 0.5
Rotasi Imej	Rotasi Imej	Memutar gambar dalam lingkungan - 45 darjah ke 45 darjah. Kebarangkalian berlakunya fungsi ini adalah 0.5.

Semua proses yang telah dinyatakan dilaksana menggunakan *Torchvision*.

4.3.3 Model yang diubahsuai

Model ini akan diubahsuai dengan tujuan untuk mengkaji perbezaan daripada segi ketepatan model selepas perubahan telah dilakukan pada algoritma UNet. Dalam proses ini, perubahan daripada segi *hyperparameter* yang digunakan dalam CNN dan struktur UNet akan dilaksanakan.

Model ini akan menjalani beberapa perubahan yang akan melibatkan beberapa *hyperparameter* dalam algoritma CNN yang digunakan dalam UNet. Jadual 4.7 menunjukkan *hyperparameter* yang diubahsuai.

Jadual 4.7: *Hyperparameter* asal dan *hyperparameter* yang diubahsuai

<i>Hyperparameter</i>	Jenis algoritma yang asal	Jenis algoritma yang diubahsuai
Algoritma <i>optimization</i>	Root Mean Squared Propagation	Adaptive Moment (ADAM)
<i>Activation Unit</i>	RELU	Leaky RELU
<i>Kernel Size</i>	3 x 3	3 x 3
<i>Pooling</i>	Max Pooling	Max Pooling
<i>Padding</i>	1	1
<i>Learning Rate</i>	0.0001	0.0001
<i>Batch Size</i>	32	32

4.4 Fasa Pengujian

Selepas fasa pembangunan, pengujian model akan dilaksanakan. Ketepatan model telah diuji mengikut nilai *dice-coefficient* atau F-Score. Ketepatan model juga akan dinilai dalam metrik *mean Intersection over Union* (mIoU). Pengujian terhadap sistem yang telah dibangunkan akan dilaksanakan untuk menentukan ketepatan model, kelemahan dan masalah yang berlaku. Pengujian terhadap GUI yang telah dibangunkan juga telah dilaksanakan. Ketepatan model yang telah dinilai akan disentuh dan dibincang dalam Bab 5.

5 HASIL KAJIAN

Selepas proses-proses pengujian telah dijalankan terhadap model yang dilatih, kita mendapati bahawa model dengan proses *data augmentation* merupakan model yang terbaik, mencapai ketepatan F-Score sebanyak 74.26% dan 47.17% dalam metrik mIoU. Bagi model tanpa proses *data augmentation*, ketepatan model mengikut mIoU adalah 46.88% manakala ketepatan model mengikut *F-Score* adalah 74.15%. Model dengan proses *data augmentation* telah mencapai ketepatan model yang terbaik bagi kedua-dua sukatan pengujian, 47.17% dan 74.26% bagi mIoU dan *F-Score* masing-masing. Akhirnya, model yang telah diubahsuai mencapai ketepatan sebanyak 73.14% dan 46.03% masing-masing. Jadual 4.8 menunjukkan keputusan model-model yang telah dibangunkan.

Jadual 4.8: Keputusan model-model yang telah dibangunkan

Model / Sukatan Pengujian	mIoU	F-Score
Model tanpa proses data augmentation	0.4688	0.7415
Model dengan proses data augmentation	0.4717	0.7426
Model yang diubahsuai	0.4603	0.7314

6 KESIMPULAN

Model ini mempunyai beberapa kelebihan. Model ini telah membuktikan bahawa model yang terlatih dengan set data UAV dan set data satelit dapat mengesan dan menghasilkan segmentasi kapal dan bot daripada pandangan mata burung dan daripada pandangan satelit pada masa yang sama. Selain itu, model yang terlatih ini dapat mengesan dan menghasilkan segmentasi untuk kapal dan bot yang kecil. Ketepatan model yang terbaik mengikut *F-Score* yang dicapai adalah 74.26%. Namun, ketepatan model yang terbaik mengikut *mean IoU* yang dicapai adalah 47.17% sahaja.

Namun, model ini terdapat beberapa kekurangan. Antaranya adalah model ini tidak boleh menerima fail dalam bentuk video. Selain itu, tiada perubahan yang ketara berlaku terhadap model ini selepas pengubahsuaian dilakukan. Contohnya, perbezaan antara model dengan proses data augmentation dengan model tanpa proses data augmentation dan model yang diubahsuai adalah 0.09% dan 1.12% sahaja. Akhirnya, model yang terlatih tidak mempunyai set data daripada UAV yang mencukupi. Hal ini boleh menjejaskan ketepatan dan kebolehan model untuk meningkatkan ketepatan model.

Bagi menambahbaikkan model ini, terdapat beberapa cadangan yang telah dicadangkan. Antaranya adalah penambahan set data. Set data yang telah digunakan adalah tidak mencukupi. Set data yang kurang mencukupi adalah set data daripada UAV. Penambahan set data daripada UAV berpotensi untuk meningkatkan ketepatan model. Selain itu, penukaran lapisan *encoder* berpotensi untuk meningkatkan ketepatan model. Contoh lapisan *encoder* adalah ResNet dan VGG. Akhirnya, Penambahan teknik-teknik *data augmentation* boleh meningkat kepelbagaian imej. Pengeksperimentasi imej *cropping* untuk menfokuskan kepada *region of interest* (ROI) boleh dijalankan.

7 RUJUKAN

- Airbus DS Geo S.A. 2018. Airbus Ship Detection Challenge. Kaggle. <https://www.kaggle.com/c/airbus-ship-detection/rules>.
- Anggiratih, E., Putra, E. A. 2019. Ship Identification on Satellite Images Using Convolutional Neural Network and Random Forest. Indonesian Journal of Computing and Cybernetics Systems. Vol. 13, No. 2, 117 – 126.
- Buslaev, A., Iglovikov, V.I., Khvedchenya, E., Parinov, A., Druzhinin, M. & Kalinin, A.A. 2020. Albumentations: Fast and Flexible Image Augmentations. Machine Learning with Python.
- Fan, Q., Chen, F., Cheng, M., Lou, S., Xiao, R., Zhang, B., Wang, C., Li, J. 2019. Ship Detection Using a Fully Convolutional Network with Compact Polarimetric SAR Images. Remote Sensing. Vol 11, No 18. 2171.
- Goodfellow, I., Bengio, Y., & Courville, A. 2016. *Deep Learning*. The MIT Press.
- Hordiuk, D., Oliinyk, I., Hnatushenko, V. & Maksymov, K. 2019. Semantic Segmentation for Ships Detection from Satellite Imagery. 2019 IEEE 39th International Conference on Electronics and Nanotechnology (ELNANO), 454 - 457.
- Karki, S., Kulkarni., S. 2021. Ship Detection and Segmentation using UNet. 2021. International Conference on Advances in Electrical, Computing, Communications and Sustainable Technologies (ICAECT). 1 – 7.

- Krizhevsky, A., Sutskever, I. & Hinton, G.E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. Conference on Neural Information Processing System (NeurIPS) 2012, 1106-1114.
- Milesi, A. 2019. Pytorch-UNet. Github repository. <https://github.com/milesial/Pytorch-UNet>
- New Straits Times. 2016, May 23. 15 kidnap attempts foiled by Esscom since 2014.
- Nie, X., Duan, M., Ding, H., Hu, B., Wong, K. E. 2020. Attention Mask R-CNN for Ship Detection and Segmentation from Remote Sensing Images. IEEE Access. 8. 9325 – 9334.
- O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G.V., Krpalkova, L., Riordan, D., Walsh, J. 2019. Deep Learning vs. Traditional Computer Vision. CVC 2019: Advances in Computer Vision, 128-144.
- Ribeiro, R., Cruz, G., Matos, J., & Bernardino, A. 2017. A Data Set for Airborne Maritime Surveillance Environments. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(9), 2720-2732.
- Ronneberger, O., Fischer, P., Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. 2015 Medical Image Computing and Computer Assisted Intervention.
- SDLC – Agile Model. (n.d). Tutorials Point. https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm [29 Oktober 2020].
- Szeliski, R. 2010. Computer Vision: Algorithms and Applications. London: Springer-Verlag London.
- The Star. 2020, January 27. 40 kidnapping cases thwarted in Sabah since 2018.
- Wang, J., Ma, Y., Zhang, L., Gao, X., Wu, D. 2018. Deep Learning for Smart Manufacturing: Methods and Applications. *Journal of Manufacturing Systems*. 48, 144-156.
- Windsor, G. 2020. 5 Stages of the Agile System. Medium. <https://medium.com/brightwork-collaborative-project-management-blog/5-stages-of-the-agile-system-development-life-cycle-brightwork-com-a207bdf61696>