MENDAPATKAN IMEJ BERASASKAN LAKARAN DENGAN DATA SKALA KELABU

LOH CHEE HUI
KOK VEN JYN

Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia

ABSTRAK

Pengambilan imej berasaskan lakaran atau Sketch-Based Image Retrieval (SBIR) ialah tugas mendapatkan semula (retrieve) imej daripada pangkalan data imej semula jadi daripada lakaran yang dilukis dengan tangan. Dalam situasi ideal, SBIR seharusnya dapat mengekstrak imej mengikut ciri lakaran seperti garisan dan bentuk. Tetapi pada masa ini, kaedah terkini kebanyakan hanya mampu mendapatkan semula imej dari kelas lakaran yang sama tetapi bukan mengikut ciri-cirinya. Selain itu, kaedah semasa melatih model dengan latihan dan ujian set data bercampur yang menyebabkan keputusan yang diskriminatif. Kertas kerja yang dirujuk dalam projek ini memperkenalkan Zero Shot Learning (ZSL) untuk SBIR, dan seterusnya, model generatif bersyarat mendalam diperkenalkan. Sementara model yang diperkenalkan mampu menghasilkan keputusan yang baik dibawah tetapan ZSL, nampanya ketepatan yang dicapai masih boleh diperbaikan. Projek ini akan ditumpukan untuk bekerja di atas model tersebut dengan menggunakan ciri yang diekstrak daripada set data skala kelabu ke atas model tersebut, dengan tujuan meningkatkan prestasi model.

1 PENGENALAN

Kebelakangan ini, sambungan internet yang makin luas dan jalur lebar yang makin tinggi telah menyumbang untuk meningkatkan bilangan pengguna internet dan pertumbuhan exponent kandungan multimedia atas talian. Secara khusus, internet dipenuhi dengan kandungan imej atas sebab kepentingannya bagi pelbagai kegunaan seperti perkongsian maklumat, komersial, penerbitan majalah, dan sebagainya. Menerusi keadaan ini, cara kami untuk mencari atau mendapatkan semula maklumat daripada internet juga perlu berkembang. Sebelum ini, cara yang biasa digunakan termasuk memberikan maklumat bentuk dalam teks (text-based image retrieval), atau terus muat naik gambar bagi tujuan mencari imej lain yang serupa (content-based image retrieval). Cara ini adalah atas motivasi sukar untuk huraikan sebahagian imej yang hendak dicari dengan menggunakan teks. Tetapi, cara ini juga menghadapi isu seperti tidak memiliki imej yang serupa untuk mencari sesuatu imej.

Berhubungan dengan itu, pencarian imej berasakan lakaran (SBIR) menjadi tumpuan penyelidikan terkini. Dalam masyarakat yang pesat membangun ini, hampir semua orang mempunyai peranti skrin sentuh, majoritinya telefon pintar, pencarian imej berasakan lakaran akan menjadi arus perdana dalam pencarian imej sebaik sahaja teknologi ini matang.

2 PENYATAAN MASALAH

Kertas penyelidikan utama yang dirujuk dalam projek ini, iaitu “A Zero Shot Framework for Sketch-based Image Retrieval” oleh Yelamarthi (1), pembelajaran pukulan sifar (ZSL) telah diperkenalkan dalam topik ini dengan tujuan menguji pencarian balik imej dengan lakaran yang tidak pernah dinampak supaya tidak menggalakkan pembelajaran khusus kelas semasa latihan. Manakala model generatif yang dicadangkan dalam penyelidikan ini iaitu pengekod variasi konvolusi (CVAE) berprestasi terbaik antara model terkini dalam SBIR, tetapi ketepatan keputusan akhir masih rendah iaitu, 0.2 hingga 0.3 sahaja. Oleh itu, ketepatan masih perlu ditingkatkan.

Yelamarthi (1) mengatakan bahawa model SBIR secara ideal harus fokus kepada mempelajari komponen imej dengan lakaran yang mempunyai ciri bentuk yang sama. Ketepatan yang rendah seharusnya diperbaiki dengan membolehkan model mempelajari ciri bentuk imej dengan lakaran.

3 OBJEKTIF KAJIAN

- Membuat perbandingan mengasaskan purata ketepatan (mAP) antara keputusan model yang dilatih dengan ciri imej skala kelabu dengan model yang dilatih dengan ciri imej berwarna.
- Mengenal pasti kesan skala warna ke atas tugasan SBIR, dengan visualisasikan fokus model pengekstrakan ciri dan juga keluaran setiap lapisan tentang imej berskala kelabu dan imej berwarna,

4 METODOLOGI KAJIAN

4.1 Reka Bentuk Set Data

Kaedah SBIR terkini, tumpuannya hanya mendapatkan imej yang terkandung dalam kelas yang sama, tetapi tidak semestinya mempunyai ciri bentuk yang sama seperti dalam lakaran. Menurut kajian Yelamarthi (1), kaedah yang sedia ada hanya belajar untuk kaitkan lakaran dengan kelas imej yang termasuk dalam latihan dan gagal untuk digeneralisasikan kepada kelas yang tidak terkandung. Dalam Jadual 4.1, di bawah tetapan ZSL, ketepatan model-model terkini tersebut semua mengalami penurunan prestasi dan telah membuktikan model-model terkini mempunyai sifat diskriminasi dan tidak dapat mengeneralisasi baik dengan kelas imej yand tidak terkandung dalam set data latihan.

Kaedah	Ketepatan		Purata ketepatan (mAP)	
	Asal	ZSL	Asal	ZSL
Siamese-1	-	0.243	-	0.134
Siamese-2	0.690	0.251	-	0.149
Coarse-grained triplet	0.761	0.169	0.518	0.083
Fine-grained triplet	-	0.155	0.573	0.081
DSH	0.886	0.153	0.783	0.059

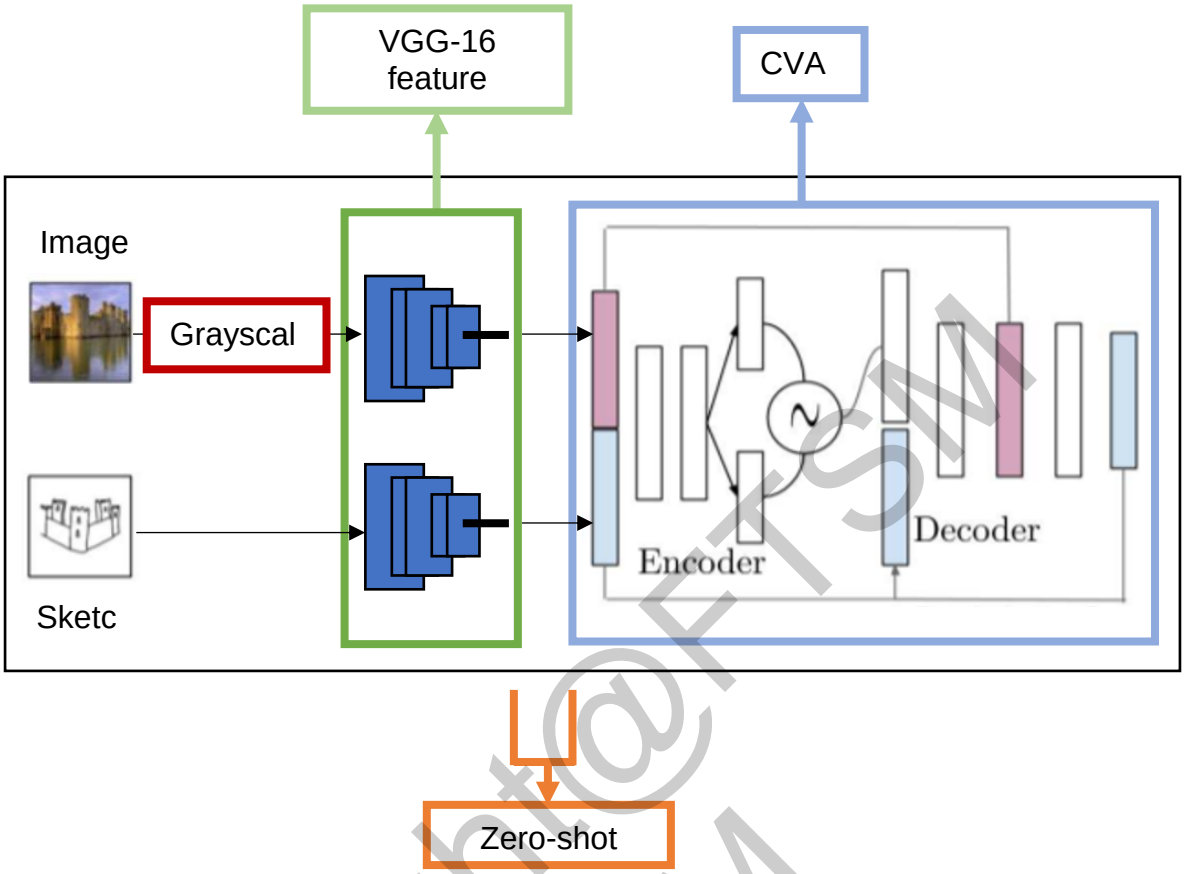
Jadual 4.1: Ketepatan dan *mean Average Precision* mAP yang dianggarkan dengan mendapatkan 200 imej oleh Yelamarthi (1)

Di bawah adalah takrifan rasmi bagi tetapan pukulan sifar dalam SBIR. Biarkan $S = \{(x_i^{\text{sketch}}, x_i^{img}, y_i) \mid y_i \in \mathcal{Y}\}$ menjadi triplets bagi lakaran, imej dan label kelas dengan $\mathcal{Y}$ ialah set semua label kelas dalam S. Kemudian, pembahagian dibuat atas label kelas dalam data $\mathcal{Y}_{\text{train}}$ dan $\mathcal{Y}_{\text{test}}$ masing-masing. Sejajar dengan itu, membiarkan $S_{tr} = \{(x_i^{sketch}, x_i^{img}) \mid y_i \in Y_{train}\}$ and $S_{te} = \{(x_i^{sketch}, x_i^{img}) \mid y_i \in Y_{test}\}$ ialah pembahagian S ke dalam set latihan dan ujian. Dengan cara ini, kami membahagikan data berpasangan ke dalam set latihan dan ujian supaya tiada satu pun lakaran daripada kelas ujian muncul dalam set latihan.

Membiarkan D menjadi pangkalan data semua imej dan gl ialah pemetaan daripada imej ke label kelas. D dibahagikan kepada $D_{tr} = \{x_i^{img} \in D \mid g_I(x_i^{img}) \in Y_{\text{train}}\}$ dan $D_{te} = \{x_i^{img} \in D \mid g_I(x_i^{img}) \in Y_{\text{test}}\}$. Model perolehan semula dalam rangka kerja ini hanya boleh dilatih pada S_{tr}. Tetapan penilaian ini memastikan bahawa model tidak boleh hanya mempelajari pemetaan daripada lakaran ke label kelas dan mendapatkan semula semua imej menggunakan maklumat label. Model kini perlu mempelajari ciri umum yang menonjol antara lakaran dan imej dan menggunakannya untuk mendapatkan semula imej untuk pertanyaan yang berasal dari kelas yang tidak kelihatan.

4.2 Struktur Keseluruhan Model

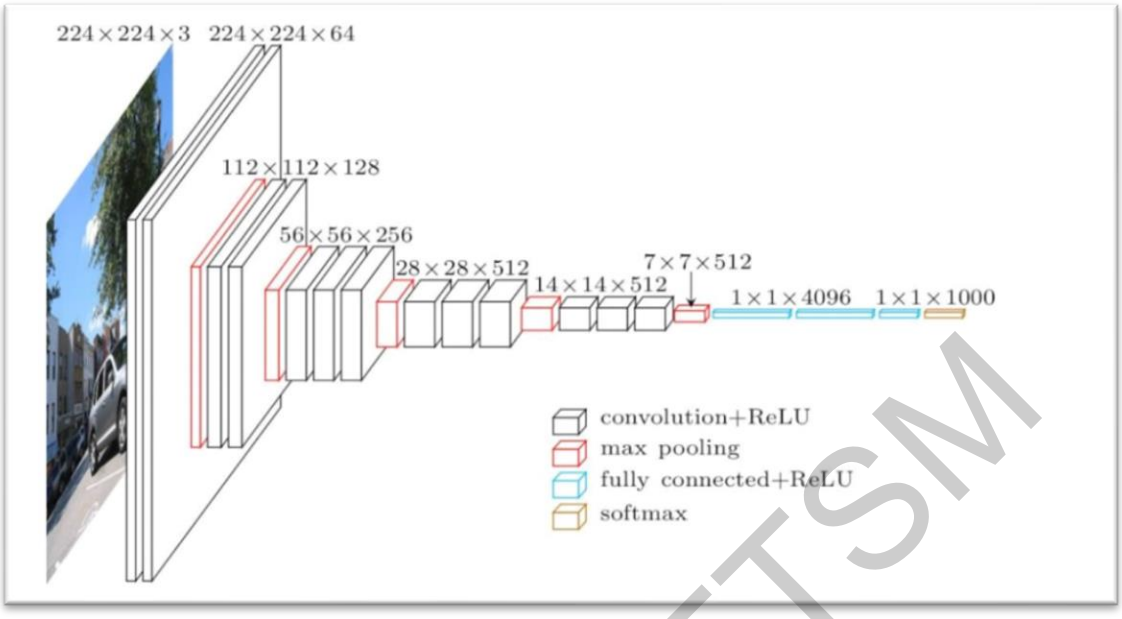
Untuk melatih model CVAE, ciri-ciri set data diekstrak dahulu dengan VGG-16 model. Model CVAE diperkenalkan dalam tugas SBIR untuk tujuan menangani masalah model sedia ada yang tidak dapat mengeneralisasi dengan kelas yang tidak dilihat semasa latihan model. CVAE dijangka dapat menjana informasi bagi pasangan lakaran dan imej melalui pengekod dan penyahkod. Dalam projek ini, untuk tujuan membandingkan prestasi model, set data imej juga akan dipindah ke skala kelabu dan model dijangka dapat belajar perhubangan pasangan imej dan lakaran lebih baik.



Rajah 4.1: Struktur keseluruhan model projek ini.

4.3 Pengekstrakan Ciri Dengan Model Vgg-16

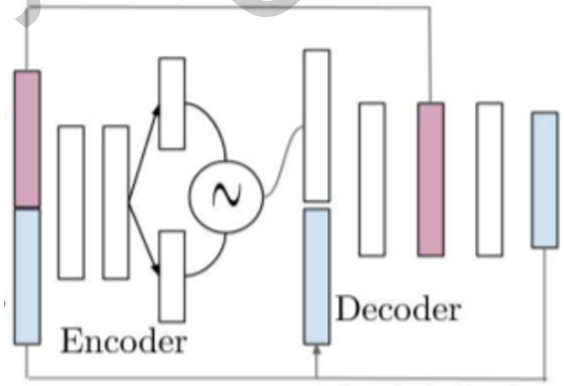
VGG-16 ialah seni bina rangkaian CNN yang digunakan dalam projek ini untuk mengekstrak ciri daripada imej dan lakaran. Seni bina VGG-16 pernah memenangi pertandingan Imagenet pada tahun 2014 dan dianggap sebagai salah satu seni bina model penglihatan yang paling cemerlang. VGG-16 juga dipanggil Oxford Net dan dinamakan daripada *Visual Geometry Group* yang mempunyai 16 lapisan. Tempat unik VGG-16 ialah, ia menumpukan pada mempunyai lapisan konvolusi penapis 3x3 dengan langkah (stride) 1 dan sentiasa menggunakan lapisan maxpool yang sama iaitu penapis 2x2 dengan langkah 2. Ia mengikut susunan ini melalui seluruh seni bina. Pada akhirnya, VGG-16 mempunyai 2 lapisan bersampung sepenuhnya diikuti dengan softmax sebagai output. Visualisasi model VGG 16 ditunjukkan dalam gambar Rajah 4.2.



Rajah 4.2: Seni bina rangkaian VGG-16 untuk pengekstrakan ciri

Sebagai maklumat tambahan untuk VGG-16, di mana ia adalah CNN. Peranan VGG-16 di ini ialah mengurangkan imej ke dalam bentuk yang lebih mudah diproses, pada masa yang sama tanpa kehilangan ciri yang penting untuk mendapatkan ramalan model yang baik. Imej masukkan dikategori dalam ruang warna seperti RGB, skala kelabu, HSV, dll. Di mana dalam projek ini, imej input asal adalah berwarna RGB, tetapi telah ditukar ke skala kelabu bagi melatih model CVAE sekali lagi dengan ciri-ciri set data skala kelabu.

4.4 Pengekod Auto Variasi Bersyarat



Rajah 4.3: Seni bina rangkaian neural autoencoder variasi bersyarat

Baru-baru ini, model generatif mendalam telah menunjukkan keupayaan yang luar biasa untuk menghasilkan kandungan yang realistic seperti imej. Secara ringkas, *Variational Autoencoder* (VAE) ialah pengekod automatik yang pengedaran pengekodannya ditetapkan semasa latihan untuk memastikan ruang terpendamnya mempunyai sifat yang baik membolehkan kami menjana data baharu. VAE berupaya untuk menjana data sintetik baharu daripada perwakilan

termampat. Bukan seperti autoencoder biasa yang hanya membina semula data yang dimasukkan, VAE boleh menjana data yang baharu dengan mendapatkan taburan kebarangkalian data dalam pembolehubah terpendam yang berbeza sebagai input.

Conditional Variational Autoencoder (CVAE) merupakan sambungan VAE, ia terdiri daripada pengekod dan penyahkod seperti pengekod auto yang lain, dimana pengekod di sini kadang kala dirujuk sebagai model pengecam dan penyahkod dirujuk sebagai model generatif. CVAE menjalankan proses pembinaan semula dengan membina model pengekod untuk mengeluarkan julat nilai yang mungkin, dan daripadanya, akan sampel secara rawak untuk dimasukkan ke dalam model penyahkod bersama-sama dengan "syarat" sebagai panduan kepada keluaran yang dijangka. Dalam projek ini, dimana input kepada pengekod adalah pasangan ciri-ciri lakaran dan imej, julat nilai yang mungkin adalah informasi yang dipelajari daripada pasangan tersebut. Dengan ini, imej yang dihasilkan dengan ciri lakaran sebagai syarat, dan sampel daripada keluaran rawak pengekod dapat dihasilkan. Dari sini model menjangkakan prestasi model itu mendapat balik imej dengan mengira jarak kosinus antara imej yang dihasilkan dengan kumpulan kelas imej yang sebenar.

CVAE memetakan taburan terdahulu pada pembolehubah pendam tersembunyi $p(z)$ kepada taburan data $p(x)$. $p(z|x)$ posterior yang sukar dianggarkan oleh taburan variasi $q(z|x)$yang diandaikan sebagai Gaussian dalam kertas rujukan peojek ini. Parameter taburan variasi dianggarkan daripada x melalui pengekod yang merupakan rangkaian neural yang diparameterkan oleh ϕ. Taburan bersyarat $p(x|z)$ dimodelkan oleh rangkaian penyahkod yang diparameterkan oleh θ. Variasi untuk px boleh ditulis sebagai:

$$p(x) \geq \mathcal{L}(\phi, \theta; x)$$
$$= -D_{KL}\big(q_\phi(z \mid x) \parallel p_\theta(z)\big) + \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x \mid z)]$$

Begitu juga, adalah mungkin untuk memodelkan kebarangkalian bersyarat $p(x|y)$. Taburan dimodelkan kebarangkalian ke atas imej yang dikondisikan pada lakaran seperti $P\big(x_{img} \mid x_{sketch}\big)$. Ikatan kini menjadi:

$$\mathcal{L}\big(\phi, \theta; x_{img}, x_{sketch}\big) =$$
$$-D_{KL}\left(q_\phi\big(z \mid x_{img}, x_{sketch}\big) \parallel p_\theta(z \mid x_{sketch})\right) +$$
$$\mathbb{E}\big[\log p_\theta\big(x_{img} \mid z, x_{sketch}\big)\big]$$

Tambahan pula, untuk menggalakkan model mengekalkan penjajaran terpendam lakaran, kami menambah regularisasi pembinaan semula kepada objektif. Dalam erti kata lain, kami memaksa kebolehbinaan semula ciri lakaran daripada ciri imej yang dihasilkan melalui

rangkaian saraf satu lapisan f_{NN} dengan parameter ψ. Semua parameter $\theta, \psi \& \phi$ dilatih dari hujung ke hujung. Kehulangan regularisasi buleh dinyatakan sebagai:

$$\mathcal{L}_{\text{recons}} = \lambda . \left\| f_{NN}(\hat{x}_{img}) - x_{sketch} \right\|_2^2$$

Di sini, λ ialah parameter hiper yang perlu ditala.

Dalam latihan model CVAE, pengoptimum Adam digunakan dengan kadar pembelajaran $\alpha = 2 \times 10^{-4}, \beta_1 = 0.5, \beta_2 = 0.999$ dan saiz kelompok 128.

4.5 Model Perolehan Semula

G_θ dilatih pada pasagan ciri lakaran-imej daripada kelas yang dilihat. Semasa ujian, bahagian penyahkod rangkaian digunakan untuk menjana beberapa vector ciri imej x^I_{gen} yang dikondisikan pada lakaran ujian dengan vector pendam daripada taburan terdahulu $p(z) = \mathcal{N}(0, I)$. Untuk lakaran ujian x_S yang sepadan dengan kelas ujian, menjana set $\mathcal{I}_{x_S}$ yang terdiri daripada N (parameter hiper) seperti sample x^I_{gen} . Kemudian, mengelompokkan sampel $\mathcal{I}_{x_S}$ yang dijana ini menggunakan pengelampokan K-means dan mendapatkan pusat gugusan K $C_1, C_2, \dots, C_k$ untuk setiap lakaran ujian. Kemudian, mendapatkan 200 imej x^I_{db} daripada pangkalan data imej berdasarkan metrik jarak berikut:

$$\mathcal{D}(x_I^{db}, \mathcal{I}_{x_S}) = \min_{k=1}^{K} \text{cosine}(\theta(x_I^{db}), C_k)$$

di mana θ merupakan fungsi VGG-16. Secara empiric memerhatikan bahawa K=5 memberikan hasil terbaik untuk mendapatkan semula. Metrik jarak lain yang biasanya digunakan dalam pengelompokan telah dipertimbangkan tetapi ini memberikan hasil yang terbaik.

4.6 Model Visualisasi

Seperti yang dinyatakan dalam objektif kajian, selepas mendapatkan keputusan latihan model dengan set data skala kelabu dan dengan set data berwarna, eksperimen visualisasi akan dijalankan dengan tujuan mengenal pasti kesan imej kelabu ke atas fokus model. Sejak pengekstrakan ciri dijalankan dengan mode VGG-16 dalam projek ini, visualisasi juga dijalankan terhadap model VGG-16. Cara untuk visualisasikan fokus VGG-16 terhadap imej adalah dengan teknik *Gradient-weighted Class Activation Mapping* (Grad-CAM), yang digunakan untuk mencipta *heatmap* berdasarkan imej input tertentu dan model pralatihan rangkaian saraf konvolusi. Biasanya, Grad-CAM digunakan bersama kelas imej yand dipilih,

untuk pandukan model untuk fokus ke kawasan yang tertentu, untuk mengesan objek tertentu dalam imej, tetapi di sini, penggunaan Grad-CAM adalah mengenal pasti kawasan yang difokus oleh model VGG-16 tanpa sebarang panduan. Grad-CAM secara ringkasnya, bekerja dengan mengambil imej sebagai input, jalankan input melalui model, ambil keluaran lapisan, dan seterusnya mencari kecerunan keluaran lapisan kovolusi model yang terpilih. Dari situ, mengambil bahagian kecerunan yang menyumbang kepada ramalan supaya *heatmap* boleh ditindih dengan imej asal.

Visualisasi kedua yang digunakan dalam projek ini adalah visualisasikan keluaran lapisan konvolusi secara terus. Ini dijalankan dengan pilih lapisan konvolusi hendak divisualisasi. Bina model yang baharu yang hanya mengandungi lapisan yang dipilih. Seterusnya, imej akan dimasukkan melalui model baru yang dibina, dan akhirnya hasilkan output ciri dengan meramalkan imej input.

5 PEMBANGUNAN DAN PENGUJIAN PROJEK

5.1 Tetapan Eksperimen

Bagi menrealisasikan objektif projek ini, iaitu menggunakan set data berskala kelabu ke atas model CVAE dan seterusnya menghasilkan keputusan ketepatan bagi tujuan perbandingan set data berbeza jika tidak dapat meningkatkan prestasi. Pelbagai langkah turut perlu diambil sebelum model ini dapat meneruskan proses eksperimen projek ini. Penjejakan kod juga dilakukan sebelum sebarang kerja dimulakan, dalam alam kerja pengekodan, memahami kod kerja yang bukan asal daripada sendiri adalah amat penting, tanpa pemahaman yang cukup, pelbagai masalah akan tertimbul ketika bekerja pada kod tersebut. Pelbagai perubahan turut dilakukan kepada kod asal demi membolehkan model ini dapat berfungsi dan menjalankan kerja latihan dan pengiraan ketepatan model.

5.2 Pengekstrakan Ciri

Pada peringkat pertama, set data imej yang asal (*Sketchy*), iaitu yang dibekalkan oleh pengarang asal, tidak boleh digunakan. Set data yang dibekalkan tersebut adalah dalam format *numPy array* (npy), yang merupakan ciri yang telah diektrak melalui kod pre-step image.py. Kod dijalankan, tapi semua imej ditukar kepada mod skala kelabu sebelum ciri ekstrak. Selepas ini, model VGG-16 yang merupakan model pralatihan akan diimport untuk ekstrak ciri setiap imej. Pegeluaran model ini merupakan numPy array yang mewakili ciri setiap imej, bersama dengan ciri imej tersebut, lokasi setiap imej juga akan disimpan sebagai numPy array fail (npy)

mengikut susunan yang sepadan. Berikut adalah segmen kod dalam Rajah 5.1 adalah untuk mengekstrak ciri semua set data, iaitu dengan memuatkan model pralatihan VGG-16:

```
#Load VGG-16 as image model due to lack of proper image data
vgg_model = vgg16.VGG16(weights='imagenet', include_top=True)
#remove top layer (prediction layer)
vgg_model.layers.pop()
vgg_model.layers[-1].outbound_nodes = []
vgg_model.outputs = [vgg_model.layers[-1].output]
vgg_model.summary()
```

Rajah 5.1: VGG-16 model pralatihan yang digunakan dalam projek ini untuk mengekstrak set data ke set nombor yang mampu mengekalkan ciri dan diproses dengan lebih mudah.

Model ini kemudian akan digunakan untuk ektrak ciri semua data dalam projek ini, termasuk imej dan lakaran set data asal Sketchy, dan imej set data Sketchy yang diperbesarkan. Segmen kod yang ditunjukkan dalam Rajah 5.2 berikut adalah kaedah digunakan untuk memasukkan data sebagai input kepada model ini dan menghasilkan ciri input tersebut.:

```
#Generate vgg features for each image
BATCH_SIZE = 25
X_out = np.zeros((len(image_paths), 4096))
X_in = np.zeros((BATCH_SIZE, 224, 224, 3))
for ii in range(len(image_paths)//BATCH_SIZE):
    print ('Batch ' + str(ii) + ' in progress...')
    for jj in range(BATCH_SIZE):
        X_in[jj,:,:,:] = image.img_to_array( image.load_img(image_paths[ii*BATCH_SIZE + jj],color_mode='grayscale', target_size=(224, 224)) )
    X_in = preprocess_input(X_in)
    #preprocess input adequate images to the format that the model require
    X_out[ii*BATCH_SIZE:(ii+1)*BATCH_SIZE, :] = vgg_model.predict_on_batch(X_in)
```

Rajah 5.2: Bahagian "color_mode='grayscale'" menyatakan bahawa imej tersebut akan ditukar kepada mod warna kelabu sebelum dimasukkan ke dalam model VGG-16 tersebut.

Bagi set data lakaran pula, sepatutnya set data yang digunakan dalam projek ini adalah sama dengan set data lakaran yang asal, tetapi dalam kod latihan model, lokasi telah menjadi kunci untuk menggabungkan imej dengan lakaran sebagai pasangan, jadi, fail npy yang asal tidak dapat digunakan kerana lokasi lakaran tersebut adalah dari direktori computer pengarang. Oleh itu, kerja ekstak ciri juga dilakukan ke atas set data lakaran dan langkahnya adalah sama seperti ekstrak ciri imej yang disebut. Yang berbeza dangan ekstrak ciri imej, set data lakaran adalah amat besar yang sekurang-kurangnya 5 kali ganda dengan set data imej. Tambahan pula, platform yang digunakan adalah Google Colab, hanya memberikan masa larian maksimum 12 jam yang amat tidak mencukupi untuk ekstrak ciri set data lakaran yang mempunyai 70 ribu lakaran. Sebab ini, set data lakaran dibahagikan kepada 3 bahagian, dan setiap bahagian mengambil masa kira-kiranya 11 jam untuk ekstrak. Ketiga-tiga bahagian ini kemudian akan

digabungkan sebagai satu dalam selepas diimport le kod latihan model. Cara set lakaran ini dibahagikan kepada tiga bahagian untuk dilatih asing-asing ditunjukkan dengan segmen kod, Rajah 5.3 berikut:

Rajah 5.3: Set data diasingkan kepada tiga bahagian dan diekstrak satu demi satu

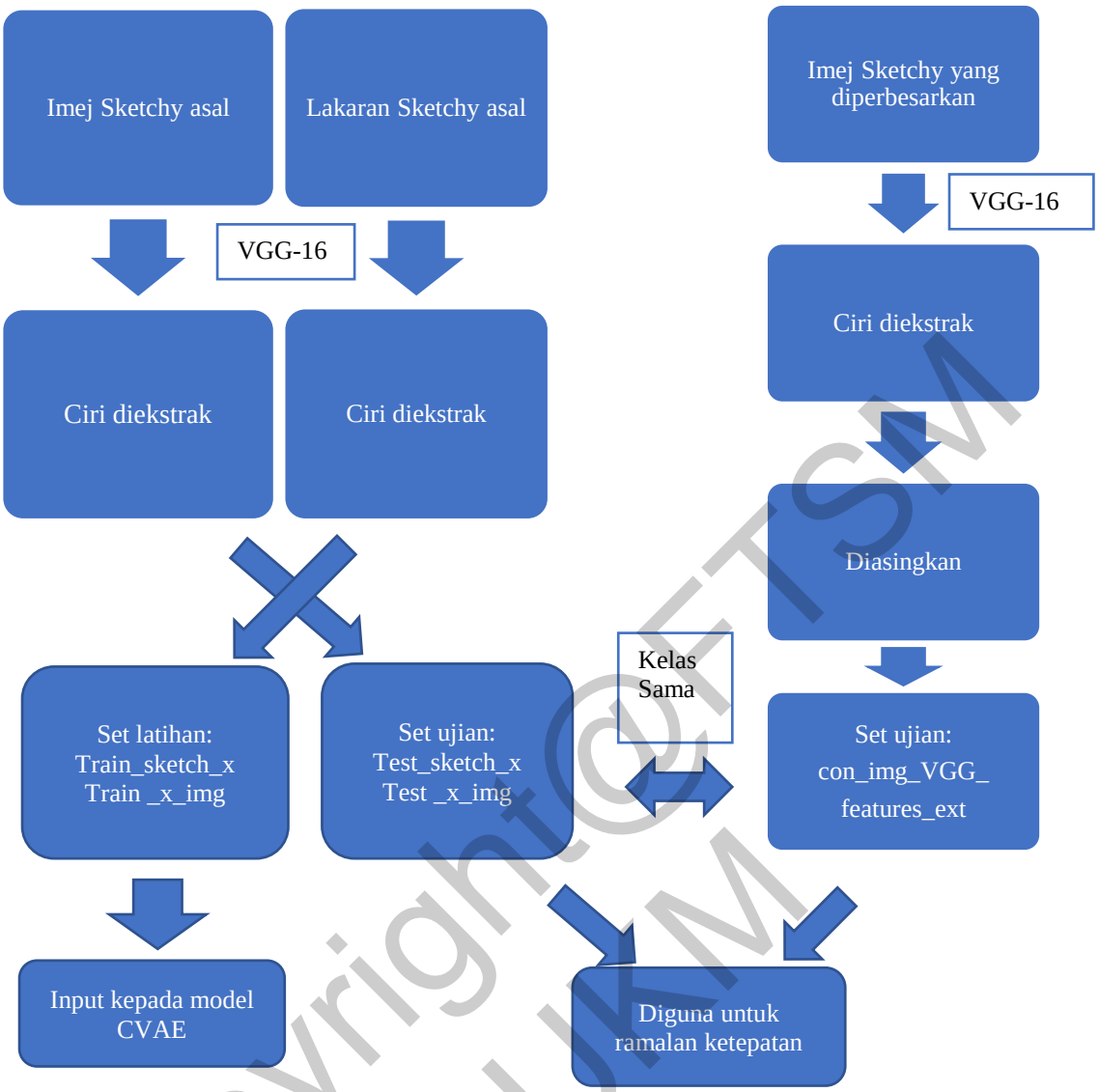
```
sketch_paths1 = np.load('/content/drive/MyDrive/Sketchy256x256/Features/sketch_paths_1.npy',mmap_mode="r")
sketch_paths2 = np.load('/content/drive/MyDrive/Sketchy256x256/Features/sketch_paths_2.npy',mmap_mode="r")
sketch_paths3 = np.load('/content/drive/MyDrive/Sketchy256x256/Features/sketch_paths_3.npy',mmap_mode="r")
sketch_VGG_features1 = np.load('/content/drive/MyDrive/Sketchy256x256/Features/vgg_sketch_features_1.npy',mmap_mode="r")
sketch_VGG_features2 = np.load('/content/drive/MyDrive/Sketchy256x256/Features/vgg_sketch_features_2.npy',mmap_mode="r")
sketch_VGG_features3 = np.load('/content/drive/MyDrive/Sketchy256x256/Features/vgg_sketch_features_3.npy',mmap_mode="r")

sketch_paths=np.concatenate((sketch_paths1, sketch_paths2, sketch_paths3))
sketch_VGG_features=np.concatenate((sketch_VGG_features1, sketch_VGG_features2, sketch_VGG_features3))
```

Rajah 5.4: Ciri yang diekstrak secara asing-asing digabungkan sebelum sambung kerja kemudian.

Satu lagi kod untuk ektrak ciri adalah prestep image ext.py. Kod ini adalah sama dengan kedua-dua kod yang disebut tadi. Kerjanya adalah untuk ektrak ciri set data *Sketchy* yang diperbesarkan oleh Liu dll dengan menggunakan perpustakaan imej *Imagenet*. Set data ini dimasukkan dalam projek ini dengan tujuan untuk memastikan model mampu mendapatkan semula kelas imej yang tidak dimasukkan dalam set data latihan.

5.3 Perasingan Set Data

Terdapat 2 set data yang digunakan dalam projek ini, iaitu Sketchy yang asal dan Sketchy yang diperbesarkan. Set data ini akan diasingkan dengan tetapan *Zero Shot.* Di mana kelas yang digunakan dalam bahagian ujian model tidak akan dimasukkan dalam set latihan. Ini adalah untuk memastikan model akhir yang dilatih dapat mengecam kelas imej yang tidak termasuk semasa laithan, dan turut dapat generalisasi dengan alam internet yang bilangan imej tidak berhenti berkembang. Aliran data dalam projek ini boleh ditunjukkan dengan Rajah 5.5 di bawah:



Rajah 5.5: Carta ini menunjukkan pengasingan set data kepada set latihan dan set ujian atau lazimnya dipanggil sebagai tetapan *Zero Shot* dalam eksperimen ini

```
trainClasses = []
for className in sketch_paths_per_class:
    if className not in test_ref_classes:
        trainClasses.append(className)
        continue
    else:
        test_sketch_paths = np.append(test_sketch_paths, sketch_paths_per_class[className])
        for test_path in sketch_paths_per_class[className]:
            train_sketch_paths.remove(test_path)
```

Rajah 5.6: Segmen kod mengasingkan kelas imej latihan dan ujian.

Test_ref_class mengandungi nama kelas imej yang akan digunakan dalam ujian. Segmen kod ini tambahkan kelas yang tiada dalam test_ref_class ke dalam *array* "trainClasses" dan tambahkan semua imej yang muncul dalam test_ref_class berserta dengan lokasinya ke dalam "test_sketch_paths".

```
#used to test, remove train classes from the extension dataset
con_image_paths_ext = []
for path in image_paths_ext:
    className = path.split(b'/')[-2]
    if className not in trainClasses:
        con_image_paths_ext.append(path)
```

Rajah 5.7: Segmen kod mengasingkan kelas imej latihan dan ujian kepada set data yang diperbesarkan dan digunakan untuk ujian.

"image_paths_ext" mengandungi semua lokasi set data Sketchy yang diperbesearkan seperti yang dinyata di atas. Segmen kod ini juga mengasingkan set data ini ke kelas latihan dengan masukkan semua lokasi dan imej yang kelasnya tidak terkandung dalam "trainClasses", iaitu kelas latihan, ke dalam "con_image_paths_ext" yang digunakan untuk kerja ujian model.

5.4 Ramalan Ketetapan

Decoder yand dilatih kemudiannya akan digunakan untuk menjana ciri imej bagi setiap lakaran ujian sebagai syarat dan pengedaran terdahulu sebagai input. Kemudian purata ciri imej akan dikira mengikut sampel rawak setiap lakaran dan dimasukkan ke dalam pengelas jiran terdekat yang dibina dengan set data imej yang diperbesarkan untuk mendapatkan indeks jarak antara ciri yang diramalkan dan imej sebenar. Akhirnya ketepatan dan mAP dianggarkan dengan mendapatkan 200 imej akan dikira berdasarkan metrik jarak diambil, dan ini ialah metrik penilaian projek ini.

6 HASIL KAJIAN

6.1 Keputusan Kuantitatif

Metrik penilaian adalah sama dengan yang asal, oleh Yelamarthi (1), menggunakan ketepatan dan purata ketepatan (mAP) dalam mendapatkan 200 imej. Ketepatan di sini merujuk kepada ketepatan bagi ambang keputusan tertentu, dan mAP merupakan purata ketepatan bagi semua ambang yang mungkin. Keputusan kuantitatif yang diperolehi dengan melatih model dengan set data yang berbeza dijadualkan dalan Jadual 6.1.

Tetapan	Ketepatan	mAP
Asal (25 *epochs*)	0.333	0.225
Skala Kelabu (15 *epochs*)	0.271	0.157
Berwarna (15 *epochs*)	0.324	0.216

Jadual 6.1: Keputusan kuantitatif dalam ketepatan dan mAP dalam mendapatkan 200 imej. Lebih tinggi ketepatan dan mAP model, lebih menunjukkan kebolehan model mendapat balik imej daripada kelas yang sama dengan lakaran yang dimasukkan.

Epoch latihan, yang berbeza digunakan dalam projek ini dengan pengarang asal. Nampaknya, model yang dilatih dengan set data skala kelabu tidak dapat berprestasi baik berbanding model yang dilatih dalam set data asal. Keputusan ini sebenarnya menunjukkan ciri yang diekstrak daripada set data kelabu dan daripada set data berwarna adalah tidak sama. Ini kerana, model yang digunakan untuk melatih dengan mendapatkan ketepatan adalah sama, oleh itu satu-satunya perbezaan ialah skala warna set data. Walau bagaimanapun, disebabkan oleh pengekangan sumber yang tersedia di *Google Colab*, latihan model tidak dapat dijalankan sepenuhnya mengikut tetapan asal.

Di bawah tetapan yang dikurangkan, hasil latihan set data kelabu hanya mampu menghasilkan 0.157 mAP. Di bawah tetapan sama, set data berwarna mampu menghasilkan mAP 0.216. Keputusan ini menunjukkan model yang dilatih dengan set data skala kelabu tidak berprestasi baik seperti model yang dilatih dengan set data berwarna. Satu siri eksperimen dijalankan untuk mengetahui bahagian mana yang mempengaruhi hasil latihan seperti ini.

6.2 Visualisasi Dengan GRAD-CAM

Visualisasi ini dilakukan dengan menggunakan Grad-CAM yang dicadangkan oleh Ramprasaath R.Selvaraju dll (2), pada model pengekstrakan, iaitu model terlatih dahulu, VGG-16. Dalam seni bina projek ini, VGG-16 akan digunakan untuk mengekstrak maklumat daripada imej dan lakaran sebelum maklumat ini digunakan sebagai input kepada pengekod

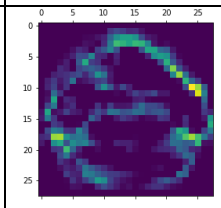
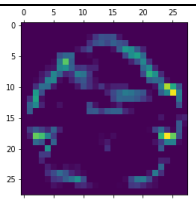
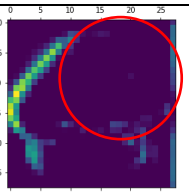
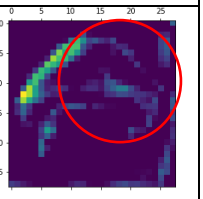
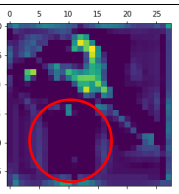
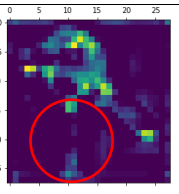
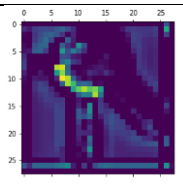
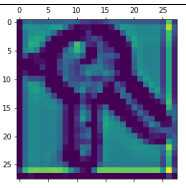
untuk membolehkan model mempelajari perkaitan antara pasangan imej-lakaran ini. Eksperimen dijalankan dengan meramalkan imej dulu, kemudian, keberatan lapisan konvolusi terakhir diambil dan divisualisasikan. Selain itu, Grad-CAM juga digunakan pada *ResNet-50* dengan input imej yang sama mengeluarkan hasil serupa, sebagai rujukan.

Eksperimen dijalankan dengan tiga kategori imej, imej ringkas, imej *Sketchy* dan imej *Sketchy* yang rumit. Imej ringkas merupakan imej yang mempunyai latar belakang yang kosong dan aspek warna yang ringkas. Imej biasa merupakan imej yang mempunyai latar belakang yang sederhana, tidak terlalu rumit yang diambil dari set data *Sketchy*. Kemudian imej Sketchy yang rumit merupakan imej yang mempunyai latar belakang yang mengelirukan, malah susah untuk mengesan objek pada pandangan pertama.

Imej ringkas	Imej *Sketchy*	Imej *Sketchy* yang rumit
		
		

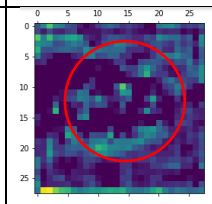
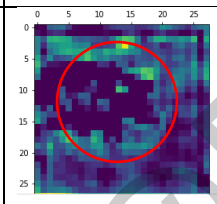
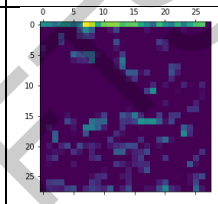
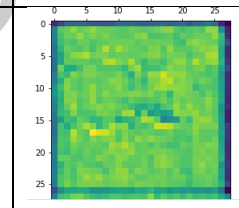
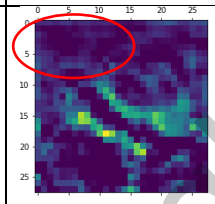
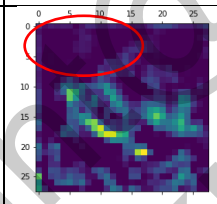
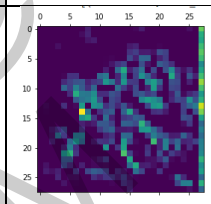
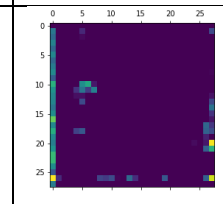
Jadual 6.2: Imej ringkas, *Sketchy* biasa dan *Sketchy* yang rumit

Bagi imej ringkas, tanpa latar belakang, hasil eksperimen ditunjukkan di Jadual 6.3 bawah.

Imej asal	Berwarna, VGG-16	Skala kelabu, VGG-16	Berwarna, ResNet-50	Skala Kelabu, ResNet-50
				
				

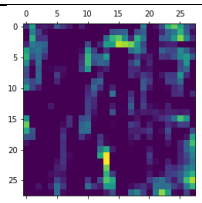
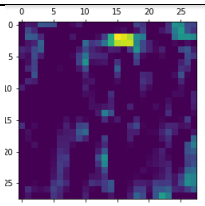
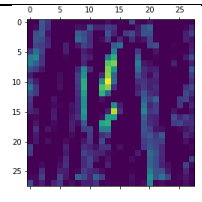
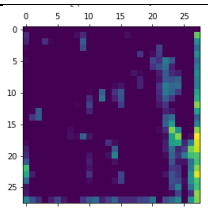
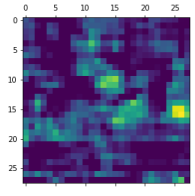
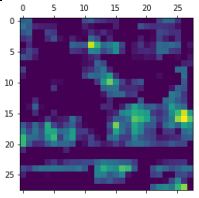
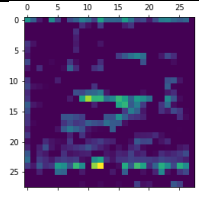
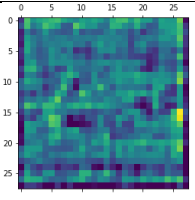
Jadual 6.3: Grad-CAM ke atas imej yang ringkas dan tiada latar belakang

Seperti yang dinyatakan oleh Yelamarthi (1), idealnya, model SBIR harus belajar kaitkan komponen lakaran dengan imej yang mempunyai ciri bentuk serupa. Dalam imej yang ringkas, tanpa gangguan informasi warna, model VGG-16 dapat fokus kepada bentuk objek. Ciri bentuk yang sebelumnya tidak difokus dalam imej yang berwarna, turut diberi perhatian, seperti yang ditunjukkan dengan bulatan warna merah. Dalam contoh yang ditunjukkan, adalah bagi imej cendawan bagi model ResNet-50, bagi imej yang berwarna, kepentingan tidak diberikan kepada bentuk, manakala imej skala kelabu dapat menangkap informasi yang tidak difokus dalam imej berwarna.

Imej asal	Berwarna, VGG-16	Skala kelabu, VGG-16	Berwarna, ResNet-50	Skala kelabu, ResNet-50
				
				

Jadual 6.4: Grad-CAM ke atas imej Sketchy dan sederhana latar belakang

Menurut Jadual 6.5, Grad-CAM digunakan ke atas imej Sketchy yang mempunyai latar belakang yang sederhana. Dengan model VGG-16, skala kelabu lebih kurang masih membantu model untuk fokus kepada bentuk objek, dan kurangkan keberatan latar belakang, seperti yang dibulatkan. Bagi ResNet-50, model mula hilang fokus dalam menggunakan imej skala kelabu.

Imej asal	Berwarna, VGG-16	Skala kelabu, VGG-16	Berwarna, ResNet-50	Skala kelabu, ResNet-50
				
				

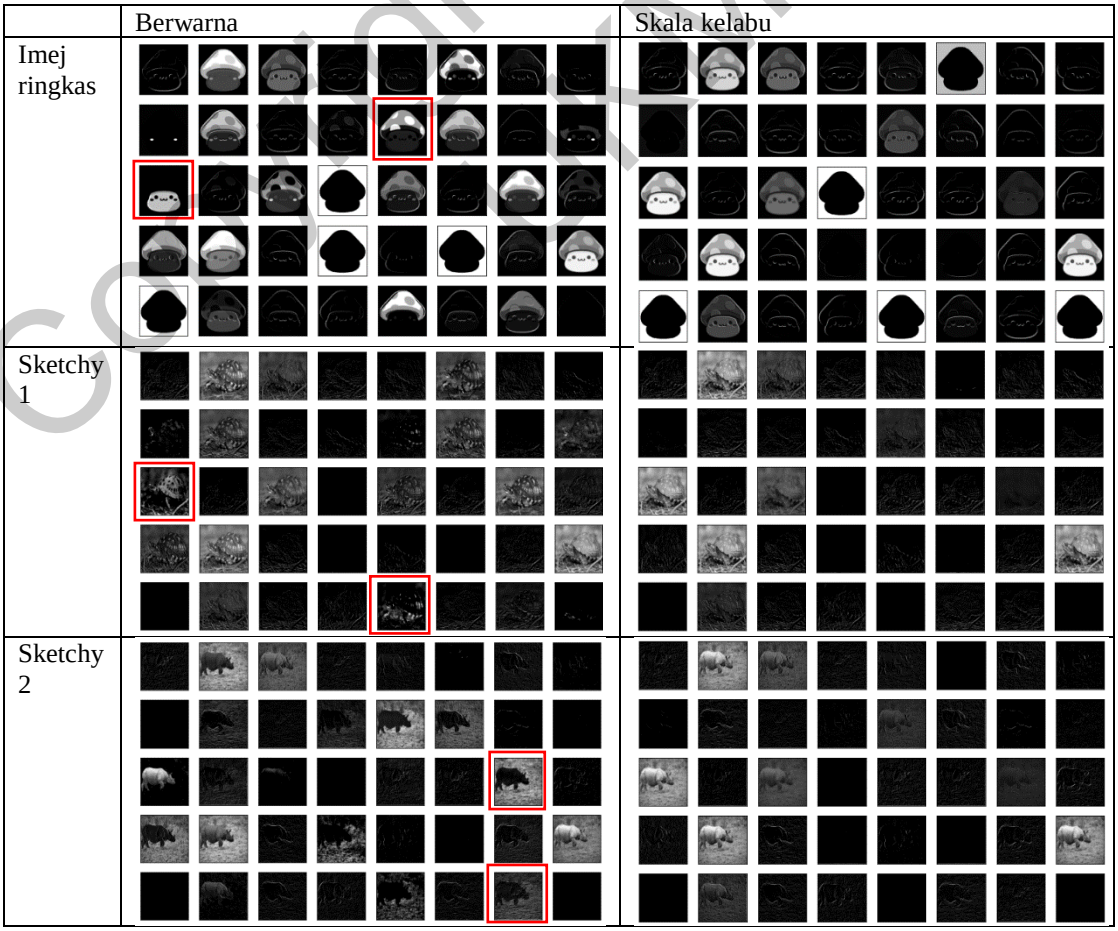
Jadual 6.5: Grad-CAM ke atas imej Sketchy dengan latar belakang yang rumit

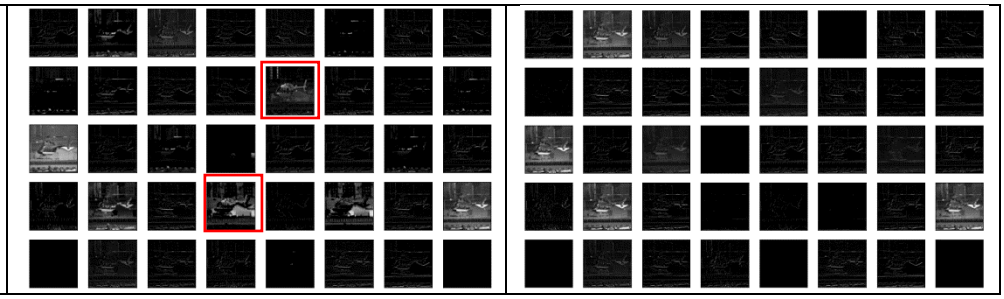
Akhirnya, Grad-CAM juga ditambah ke atas imej Sketchy yang mempunyai latar belakang yang rumit dan mengelirukan. Dari pemerhatian keluaran Grad-CAM, kedua-dua skala warna tidak dapat fokus ke atas objek dengan baik. Bagi ResNet-50, model tersebut telah hilang fokus seperti ditunjukkan dalam kolum terakhir.

Dapatan dari eksperimen ini, nampaknya skala kelabu dapat membantu model pengekstrakan ciri lebih fokus kepada bentuk objek bagi imej yang ringkas, tanpa gangguan latar belakang. Lebih banyak keberatan juga telah diberi kepada latar belakang imej ketika latar belakang imej menjadi lebih rumit. Dan set data Sketchy kebanyakan didiri daripada imej yang mempunyai jumlah kebisingan latar belakang yang sama seperti Jadual 6.3.

6.3 Visualisasi Keluaran Setiap Lapisan

Tujuan eksperimen ini adalah untuk visualisasikan secara spesifik tentang apakah ciri yang diekstrak dalam model pengekstrakan ciri dalam projek ini, VGG-16. Cara ini lebih kurang sama dengan eksperimen lepas, menggunakan konsep yang sama, dengan mengambil lapisan konvolusi untuk visualisasi apa keluarannya. Tetapi bukannya visualisasikan kawasan imej yang digunakan oleh model, eksperimen ini secara khususnya visualisasikan keluaran sebenar lapisan konvolusi.



Latar belakang rumit	

Jadual 6.6: Visualisasi keluaran lapisan konvolusi, kotak merah menunjukkan ciri yang dapat ditangkap dalam imej berwarna, tetapi tidak dari imej skala kelabu

Dari visualisasi ini, keluaran lapisan konvolusi dapat dilihat. Pada asasnya, daripada barisan imej ringkas yang ditunjukkan, dapat dilihat bahawa warna sebenarnya mampu memberikan model mengekstrak ciri yang lebih kaya, seperti topi cendawan dan badan cendawan dapat dilihat dari model imej berwarna. Tetapi pada imej skala kelabu, yang dapat dilihat hanya seluruh imej cendawan tersebut. Ini menunjukkan lebih kurang ciri diekstrak daripada imej skala kelabu. Tetapi, sebenarnya, kehilangan ciri daripada imej yang berdimensi 3 lapisan ke skala kelabu yang berdimensi dua, telah menyebabkan kurang informasi kepada model latihan untuk belajar persamaan antara pasangan imej dan lakaran dengan baik. Secara visual, objek itu lebih mirip dengan latar belakang yang sudah menjadikan model sukar untuk mengekstrak bentuk objek dalam imej. Jadi jika informasi yang output dari model pengekstrakan ciri yang kurang, dimasukkan ke dalam model latihan, sudah tentu model tidak dapat belajar untuk mengaitkan pasangan imej dan lakaran tersebut dengan baik, dan seterusnya mempengaruhi ketepatan.

6.4 Arah Penambahbaikan Masa Hadapan

Dari eksperimen visualisasi, dapat dilihat bahawa output lapisan konvolusi banyak dipengaruhi oleh latar belakang, dalam kedua-dua set data berwarna dan juga set data berskala kelabu. Dalam kes ini, modul perhatian boleh digunakan untuk model pengekstrakan ciri untuk mengurangkan ciri latar belakang yang mempengaruhi keputusan. Dalam penyelidikan lepas, modul perhatian dibuktikan oleh Kong ZM, dll, mampu membolehkan model menumpu perhatian kepada latar depan, dan menghasilkan ketepatan yang mengatasi prestasi model tanpa modul ini (3). Modul perhatian terdiri daripada lapisan konvolusi 2D yang ringkas, perceptron berbilang lapisan dan fungsi sigmoid, dengan tujuan untuk menjana lapisan untuk peta ciri input. Modul perhatian bekerja dengan konsep mengambil sub kawasan dan konteks sebagai input, dan output keberatan purata aritmetik bagi kawasan ini. Ia digunakan untuk menjadikan rangkaian saraf lebih fokus pada maklumat penting, seperti ciri bentuk objek dalam projek ini, dan bukannya mempelajari maklumat latar belakang yang tidak berkaitan.

Satu lagi arah penambahbaikan masa hadapan adalah penapisan *Gaussian* dalam pemprosesan imej, ialah penapis laluan rendah untuk mengurangkan hingar dan mengaburkan kawasan imej. Ideanya adalah untuk mengurangkan butiran dan hingaran, serupa dengan penapis min, tetapi menggunakan kernel berbeza yang bentukknya serupa bentuk gaussian, atau berbentuk loceng. Penapis *Gaussian* digunnakan dalam meredamkan hingar dalam imej dengan banyak bunyi, tetapi tidak berfungsi dengan baik dalam imej dengan kurang hingar, namun, dalam projek ini, imej dalam set data Sketchy kebanyakannya mengandungi banyak hingaran. Dengan ini, boleh mengaburkan dan melembutkan latar belakang, dan membolehkan objek menonjol dengan lebih jelas.

7 KESIMPULAN

Dalam pemprosesan imej, skala kelabu dapat membantu fokus pada bentuk bagi imej yang ringkas, tetapi apabila latar belakang menjadi lebih hingar, kedua-dua set data skala kelabu dan asal kurang fokus kepada objek dalam imej. Dalam kebanyakan kes, objek dikesan oleh bentuk, dalam projek ini, skala kelabu tidak begitu sesuai digunakan sejak dari eksperimen dapat dilihat kurangnya ciri yang dapat diekstrak dari imej, seterusnya menyebabkan model tidak dapat belajar dengan baik antara persamaan imej dan lakaran. Warna digunakan untuk menambah nilai, membolehkan model latihan ekstrak ciri yang lebih kaya, membantu model belajar kaitan imej dengan lakaran dengan lebih baik. Pada akhirnya, ciri-ciri akan kemudian akan dijadikan sebagai input pasangan imej dan lakaran, untuk model mempelajari perkaitan antara mereka, yang dimana, lebih banyak ciri, lebih baik. Seperti dalam eksperimen yang dijalankan, model yang dilatih dengan ciri imej berwarna mampu mengeluarkan mAP 0.193 manakala imej skala kelabu hanya 0.157.

Cara untuk mengurangkan berat latar belakang seperti modul perhatian, atau penapis gaussian boleh ditimbangkan untuk digunakan sebagai penambahbaikan masa hadapan. Skala kelabu pula, walaupun dapat fokus kepada bentuk objek, tetapi mempertimbangkan ciri diekstrak kena digunakan kepda model latihan untuk belajar persamaan imej dengan lakaran, ciri yang kaya kena diambil lebih penting berbandingan apa yang boleh diekstrak dengan skala kelabu.

8 RUJUKAN

(1) Yelamarthi.S.K. September 2018. "A Zero-Shot Framework for Sketch-based Image Retrieval", The European Conference on Computer Vision (ECCV), Available: https://arxiv.org/abs/1807.11724

(2) Funt, Brian. 2018. "Does Colour Really Matter? Evaluation via Object Classification"Available:https://www2.cs.sfu.ca/~funt/Funt_Zhu_DoesColourMatter_CIC26_2018.pdf

(3) D.J.Bora. February 2017. "A Novel Approach for Color Image Edge Detection Using Multidirectional Sobel Filter on HSV Color Space". International Journal of Computer Sciences and Engineering. Available: (PDF) A Novel Approach for Color Image Edge Detection Using Multidirectional Sobel Filter on HSV Color Space (researchgate.net)

(4) Hadsell, R., Chopra, S., LeCun, Y. 2006. "Dimensionality reduction by learning an invariant mapping.". IEEE Computer Society. Available: http://dblp.uni-trier.de/db/conf/cvpr/cvpr2006-2.html#HadsellCL06

(5) Liu, L., Shen, F., Shen, Y., Liu, X., Shao, L. 2017. "Deep sketch hashing: Fast freehand sketch-based image retrieval." CoRR. Available: http://dblp.uni-trier.de/db/journals/corr/corr1703.html#LiuSSLS17

(6) Yu, Q., Yang, Y., Liu, F., Song, Y.Z., Xiang, T., Hospedales, T.M.: Sketch-a-net: A deep neural network that beats humans. International Journal of Computer Vision 122(3), 411–425 (2017), http://dblp.uni-trier.de/db/journals/ijcv/ijcv122.html#YuYLSXH17

(7) Sangkloy.P , Burnell.N. July 2016. "The sketchy database". Semantic Scholar Available: [PDF] The sketchy database | Semantic Scholar

(8) fchollet. April 2020. "Grad-CAM class activiation visualization". Keras. Available: https://keras.io/examples/vision/grad_cam/#the-gradcam-algorithm

(9) Ramprasaath R. Selvaraju. Et.al December 2019. “Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization”, Institute of Electrical and Electronics Engineers Available: https://arxiv.org/pdf/1610.02391.pdf

(10) Z. Kong, Z. Fu, F. Xiong and C. Zhang, 2022. "Foreground Feature Attention Module Based on Unsupervised Saliency Detector for Few-Shot Learning," in *IEEE Access*, vol. 9, Available: Foreground Feature Attention Module Based on Unsupervised Saliency Detector for Few-Shot Learning | IEEE Journals & Magazine | IEEE Xplore

Loh Chee Hui (A176103)
Kok Ven Jyn
Fakulti Teknologi & Sains Maklumat,
Universiti Kebangsaan Malaysia