

**KEFAHAMAN KUMPULAN ORANG RAMAI DENGAN
ALGORITMA PERKELOMPOKAN PUNCAK KETUMPATAN**

VAN TZE SHAN

A161065

FAKULTI TEKNOLOGI DAN SAINS MAKLUMAT

UNIVERSITI KEBANGSAAN MALAYSIA

BANGI

2019

PENGAKUAN

Saya akui karya ini adalah hasil kerja saya sendiri kecuali nukilan dan ringkasan yang tiap-tiap satunya telah saya jelaskan sumbernya.

21 May 2019

VAN TZE SHAN

A161065

PENGHARGAAN

Setinggi-tinggi penghargaan saya rakamkan kepada penyelia saya iaitu Dr. Hadi Affendy Dahlan diatas segala tunjuk ajar dan bantuan yang diberikan kepada saya sepanjang saya menyiapkan kajian ini. Segala jasa, ilmu dan tunjuk ajar beliau akan saya kenang sepanjang hayat. Saya juga ingin mengucapkan ribuan terima kasih kepada bekas penyelia saya, Dr. Kok Ven Jyn kerana telah memberi galakkan kepada saya dan ajarkan saya cara menyelesaikan penghalang. Beliau sentiasa memberi dorongan tentang kepentingan pengurusan masa sendiri.

Tidak dilupakan juga sebuah penghargaan saya tujukan kepada keluarga tercinta saya terutamanya ibu bapa saya yang sentiasa memberikan semangat dan kekuatan sepanjang tempoh saya melengkapkan projek tahun akhir ini.

Akhir kata, terima kasih kepada semua yang terlibat secara langsung atau tidak langsung dalam menghasilkan projek tahun akhir ini. Saya juga ingin memohon ribuan kemaafan sekiranya terdapat kesilapan sepanjang menjalankan projek tahun akhir ini.

Sekian, terima kasih.

ABSTRAK

Kefahaman orang ramai adalah teknik yang penting untuk mengekalkan keselamatan awam dan melaksanakan kawalan orang ramai. Terlalu tumpat di sesuatu tempat boleh menyebabkan banyak kematian dan kecederaan apabila berlaku rempuhan manusia. Semua kemalangan ini boleh dielakkan apabila kumpulan orang ramai dikesan terlalu ramai di suatu tempat. Dalam kajian ini, saya telah menggunakan dalam algoritma perkelompokan puncak ketumpatan (*Density Peak Clustering*) untuk mendapatkan ketumpatan kumpulan orang ramai. Tujuan utama kajian ini adalah untuk mengesan kawasan yang “mosh – pit” daripada orang ramai. Kawasan “*mosh – pit*” tumpat merupakan suate kawasan yang mana orang di dalam kawasan tersebut menghuni kurang daripada 0.50 meter persegi ruang. Namun begitu, masalah utama dengan tajuk ini adalah keperluan bagi menubuhkan persekitaran untuk menjalankan kod yang diperlukan adalah sedikit tinggi bagi persembahan grafik. Selain itu, program-program yang dikendalikan adalah sukar untuk menyambung antara satu sama. Imej yang digunakan mesti adalah satu imej orang ramai yang berdefinisi tinggi. Dataset yang akan digunakan dalam kajian adalah dataset yang dipanggil ShanghaiTech Dataset kerana dataset ini mengandungi kira-kira 1200 orang ramai imej dan kira-kira 330000 kepala dilabelkan dalam imej. Kawasan orang ramai yang “mosh-pit” tumpat akan dikesan melalui pengiraan ketumpatan melalui algoritma perkelompokan puncak ketumpatan. Output projek ini ialah nilai ketumpatan orang ramai dan senarai koordinat dan kawasan orang ramai yang lebih tumpat akan dijana atas imej orang ramai tersebut.

ABSTRACT

Crowd understanding is an important technique to maintain the security of the public and perform crowd control. Too crowded in a place may cause countless dead and wound when human stampede happens. All these accidents may prevent when the crowded place is detected. In this study, I had used the I had implement the Density Peak Clustering (*DPC*) for calculating the density of the crowd. The main purpose of this study is to detect the “mosh-pit” part of the crowd. A “mosh-pit” part is when the people inside that range only occupied less than 0.50 square feet space. But the main problems when dealing with this title is the requirement for setting up the environment is slightly high for graphics rendering. The programs all run standalone is hard to connect to each other. The crowded image must be a High-Definition image for crowd estimation. The dataset will be used in this study is a dataset called ShanghaiTech Dataset because this dataset contains about 1200 crowd images and about 330000 labelled head counts in the images. The “mosh – pit” part of the crowd will be detected through the density calculation through DPC. The output of this project will be the density of crowd and list of the coordinates for the “mosh – pit” part or the denser part of the crowd and it will be marked on the crowd image.

ISI KANDUNGAN

PENGAKUAN	ii
PENGHARGAAN	iii
ABSTRAK	iv
ABSTRACT	v
SENARAI JADUAL	ix
SENARAI ILUSTRASI	x
BAB I Pendahuluan	1
1.1 Pengenalan	1
1.2 Pernyataalalahan	4
1.3 Cadangan penyelesaian	5
1.4 Objektif	6
1.5 Skop	6
1.6 Kekangan	6
1.7 Metodologi	7
1.8 Jadual	8
1.8 Kesimpulan	10
BAB II Kajian kesusasteraan	11
2.1 Pengenalan	11
2.2 Latar belakang kajian	11
2.3 Gambaran keseluruhan sistem	12
2.3.1 Pra pemprosesan	12
2.3.2 Pengambilan data nilai perspektif dan nilai GT(<i>Ground-Truth</i>)	12
2.3.3 Pengkelompok puncak ketumpatan (<i>Density peak clustering</i>)	12
2.4 Kaedah yang pernah digunakan untuk anggaran orang ramai	13

2.4.1	Kaedah berasaskan pengesanan (<i>Detection-based method</i>)	13
2.4.2	Trajektori kelompok (<i>Cluster trajectories</i>)	13
2.4.3	Regresi berasaskan ciri-ciri (<i>Feature-based regression</i>)	13
2.4.4	Rangkaian neural multiskala (<i>Multiscale neural network</i>)	14
2.4.5	Rangkaian neural konvolusi kesedaran perspektif (<i>Perspective aware convolutional neural network</i>)	14
2.4.6	Perbandingan kaedah yang sedia ada	15
2.5	Perkelompokan	16
2.5.1	Perkelompokan K-Means	16
2.5.2	Perkelompokan DBSCAN	17
2.5.3	Perkelompokan puncak ketumpatan	18
2.6	Kesimpulan	19
BAB III	Syarat – syarat spesifikasi	20
3.1	Pengenalan	20
3.2	Definisi keperluan sistem	22
3.3	Model system	23
3.4	Kesimpulan	25
BAB IV	Spesifikasi reka bentuk	26
4.1	Pengenalan	26
4.2	Reka bentuk seni bina	27
4.3	Reka bentuk algoritma	29
4.4	Kesimpulan	31
BAB V	Pembangunan sistem dan pengujian	32
5.1	Pengenalan	32
5.2	Pembangunan sistem	32

5.2.1	Segmen kod kritikal	32
5.2.2	Kod untuk menambah tugas	33
5.3	Pelan pengujian	38
5.3.1	Pengujian bukan kefungsi	38
5.3.2	Spesifikasi kes pengujian	39
5.3.3	Spesifikasi prosedur pengujian	39
5.3.4	Log Pengujian	40
5.4	Kesimpulan	57
BAB VI	Kesimpulan	58
BAB VII	Rujukan	59

SENARAI JADUAL

No. Jadual		Halaman
Jadual 1.1	Jadual untuk Semester 1	8
Jadual 1.2	Jadual untuk Semester 2	9
Jadual 2.4.6	Perbandingan algoritma sedia ada	15
Jadual 5.2.2.1	Pengiraan jarak	35
Jadual 5.3.2	Kes – kes pengujian	39
Jadual 5.3.3	Ujian masa kekuatan	39

SENARAI ILUSTRASI

No. Rajah		Halaman
Rajah 3.3.1	Seni bina projek	24
Rajah 4.2.1	Reka bentuk seni bina	27
Rajah 5.2.1.1	Kod utama	32
Rajah 5.2.2.1	Kod untuk distanceNorm()	33
Rajah 5.2.2.2	Kod untuk chi()	33
Rajah 5.2.2.3	Kod untuk fit() – Bahagian 1(a)	34
Rajah 5.2.2.4	Kod untuk fit() – Bahagian 1(b)	34
Rajah 5.2.2.5	Kod untuk fit() – Bahagian 2	35
Rajah 5.2.2.6	Kod untuk fit() – Bahagian 3	36
Rajah 5.2.2.7	Kod untuk Gtmatload()	36
Rajah 5.2.2.8	Kod untuk imgShow()	37
Rajah 5.3.4.1	Keputusan <i>Static Code Analysis</i> – Bahagian 1	40
Rajah 5.3.4.2	Keputusan <i>Static Code Analysis</i> – Bahagian 2	40
Rajah 5.3.4.3	Ujian prestasi bagi kod	52
Rajah 5.3.4.4	Keterangan dengan ujian prestasi	52
Rajah 5.3.4.5	Ujian prestasi bagi kod untuk folder Test A	53
Rajah 5.3.4.6	Ujian prestasi bagi kod untuk folder Train A	54
Rajah 5.3.4.7	Ujian prestasi bagi kod untuk folder Test B	55
Rajah 5.3.4.8	Ujian prestasi bagi kod untuk folder Train B	56s

BAB I

PENDAHULUAN

1.1 PENGENALAN

Menurut Kimballyoung "orang ramai adalah majlis perjumpaan besar bilangan orang di seluruh pusat atau titik perhatian umum". (*Deeksha S 2016*) Manakala anggaran orang ramai adalah teknik yang penting untuk pengawalan orang ramai dan mengekalkan keselamatan awam. Kesesakan manusia di satu tempat boleh menyebabkan banyak kematian dan kecederaan apabila berlaku rempuhan manusia. Sebagai contoh, pada tahun 2014, sebanyak 16 hingga 24 terbunuh dan lebih kurang 119 orang tercedera Nigeria kerana kecuaiannya Imigresen Arab Saudi. (*Okoli, Al Chukwuma, and Nnorom, Kingsley C 2014*) Satu lagi kes adalah berlaku di Duisburg, Jerman. Pesta muzik tekno Love Parade. 21 mangsa meninggal dunia dan lebih 500 kecederaan apabila rempuhan manusia berlaku di kawasan tersebut. (*D. Helbing and P. Muckerji 2012*) Semua kemalangan ini mungkin dapat dihalang apabila orang ramai dikesan terlalu ramai di suatu tempat.

Kerja-kerja yang berkaitan. Banyak algoritma yang telah dicadangkan tentang anggaran kos orang ramai. Terdapat kaedah berasaskan pengesanan untuk mengesan jumlah kepala dalam orang ramai (*P. Viola, M. J. Jones, and D. Snow 2005*). Walau bagaimanapun, had yang bagi kaedah ini adalah apabila ada orang ramai yang sangat padat, ia akan memberi kesan kepada prestasi pengesanan. Justeru ia turut menjejaskan ketepatan.

Ada juga cadangan yang menganggarkan orang ramai dengan menggunakan berasaskan ciri-ciri regresi yang kumpul latar hadapan dan meraih ciri-ciri daripada latar hadapan dan seterusnya menggunakan regresi yang berfungsi untuk anggaran orang ramai. (*A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos 2008*), (*A. B. Chan and N. Vasconcelos 2009*)

Suatu algoritma yang menarik tentang anggaran orang ramai adalah dicadangkan oleh Miaoqing Shi (*Miaoqing Shi, Zhaohui Yang, Chao Xu, Qijun Chen 2018*), yang menggunakan rangkaian neuron konvolusi perspektif – sedar untuk

pengiraan kumpulan orang ramai. Dalam cadangan mereka, mereka mula-mula memperkenalkan peta perspektif kebenaran tanah bagi melatih rangkaian. Ia adalah wajar sebagai fungsi linear paksi menegak imej dengan menggunakan geometri projekatif. Fungsi linear ini adalah berdasarkan nilai perspektif yang dikira daripada piksel sampel. Mereka meregresi peta ketumpatan dengan rangkaian tulang-belakang tunggal dengan dua skala yang berbeza. Rangkaian yang berpenapis kecil (cth: 3×3) menggabungkan kedua-dua output ketumpatan melalui lapisan perspektif sedar berberat. Pemberat belajar melalui peta perspektif ramal yang pengubahan tak linear dalam rangkaian. Output akhir adalah kukuh dengan perspektif yang berbeza-beza dan skala. Mereka mengira kesimpulan ketumpatan kumpulan orang ramai secara terus melalui imej keseluruhan.

Urusan sistem orang ramai mudah alih telah dicadangkan pada tahun 2017 oleh IJARCCE. (*Waffa M. Shalash, Aliaa Al Hazimi, Basma Al Zahrani 2017*) Dalam cadangan mereka, mereka telah menyatakan bahawa rempuhan manusia biasanya berlaku di orang ramai statik yang tumpat. Oleh yang demikian, sistem tersebut mengesan jenis orang ramai jika ada orang ramai statik yang tumpat. Pengguna akan dimaklumkan apabila mereka berada di orang ramai statik yang tumpat. Sistem mengesan ketumpatan orang ramai dengan kamera IP.

Terdapat cadangan yang memperkenalkan jenis orang ramai dan cara mudah untuk mengenal pasti jenis orang ramai daripada Dillion Cariveau. Dia bercadang untuk mengira orang ramai dengan mengenalpasti berapa kaki persegi ruang satu orang menduduki. Bagi jenis pertama orang ramai, orang ramai statik, di orang ramai yang longgar, seseorang akan menduduki 10 kaki persegi ruang manakala dalam orang ramai yang lebih padat, seseorang biasanya menduduki 4.5 kaki persegi ruang, dan dalam orang ramai yang berketumpatan tinggi, seseorang hanya menduduki 2.5 kaki persegi ruang. Jenis kedua orang ramai adalah orang ramai mudah alih. Jenis orang ramai ini adalah lebih sukar untuk menjalankan anggaran kerana orang memasuki dan meninggalkan orang ramai tersebut dari masa ke semasa melalui banyak tempat. (*Dillion Cariveau 2015*)

Pada tahun 1960, seorang Profesor dari bidang kewartawanan Herbert Jacobsen di Universiti Berkeley melihat pelajar-pelajar berkumpul di plaza di bawah. Konkrit di plaza adalah dalam format grid yang bahkan. Jadi beliau dapat mengira jumlah pelajar

dalam pemilihan petak grid dan didarabkan dengan jumlah petak di plaza. Daripada ini, beliau telah berasal peraturan ketumpatan yang asas adalah seperti berikut: Dalam kumpulan yang kurang tumpat, 1 orang menduduki setiap 1.5 meter persegi ruang, manakala bagi kumpulan yang ketumpatan sederhana, 1 orang adalah menduduki 1 meter persegi ruang. Bagi kumpulan yang berketumpatan lebih tumpat, 1 orang akan menduduki 0.5 meter persegi ruang. Bagi kawasan “mosh-pit” tumpat, 1 orang menduduki setiap 0.25 meter persegi ruang. (*MA Security Group 2017*)

1.2 PERNYATAAN MASALAH

Untuk mengelakkan rempuhan manusia berlaku apabila orang ramai terlalu tumpat dan mengurangkan jumlah kecederaan dan kematian, kita perlu mengambil beberapa tindakan keselamatan.

"Pada ketumpatan rendah, apabila semua orang boleh bergerak bebas, pergerakan orang ramai akan dinamik seperti gas," katanya. "Apabila ketumpatan naik, maka akhirnya pergerakan orang ramai akan dikekang dan ia menjadi lebih seperti cecair. Apabila ketumpatan amat tinggi, orang ramai akan dihipit antara badan-badan lain, ianya lebih seperti bahan butiran." Seperti pasir, atau beras, atau kecil pebbles. Berkata penyelidikan oleh Helbing. (*Helbing 2012*)

Kefahaman kumpulan orang ramai adalah penting untuk dikenali kerana rempuhan manusia biasanya berlaku di orang ramai statik yang tumpat. Terdapat dua jenis sifat orang ramai, iaitu mobs dan penonton. (*Funda Durupinar, Uğur Gökükbay, Aytek Aman, Norman I. Badler 2015*) Orang ramai penonton adalah orang ramai yang pasif. Orang berkumpul dan mungkin akan bersemuka dan bersentuh bahu ke bahu. Namun begitu, tiada pergerakan perhubungan rapat yang ditemui pada orang ramai penonton. Namun begitu, kehairahan orang ramai mengubah tindakan orang ramai pada bila-bila masa. Ini berlaku apabila orang ramai mula beremosi kerana keinginan ruang dan ada gangguan lain. Penghancuran orang ramai atau rempuhan manusia biasanya berlaku di orang ramai mobs di mana tidak ada tempat keluar yang mencukupi untuk membebaskan diri dari orang ramai. Oleh itu, kita perlu memberi perhatian apabila orang ramai bertukar menjadi orang ramai yang aktif.

1.3 CADANGAN PENYELESAIAN

Anggaran orang ramai dalam penyelidikan terkini adalah dengan menggunakan rangkaian neural multiskala. Tetapi tiada maklumat perspektif dikesan atau diberikan jika menggunakan rangkaian neural multiskala. Maklumat perspektif digunakan untuk mengira kumpulan orang ramai tersebut adalah tumpat secara waktu sebenar.

Dalam projek ini, saya akan menggunakan maklumat perspektif yang didapati daripada PACNN(*Perspective Aware Convolutional Neural Network*) untuk mengira ketumpatan orang ramai. Ketumpatan orang ramai adalah meter persegi dibahagikan oleh jumlah bilangan orang dalam kawasan tersebut. Maklumat perspektif akan digunakan untuk mengira jarak antara satu titik ke lain. Kawasan yang bertumpatan tinggi seperti kawasan orang ramai “mosh-pit” akan ditandakan dengan titik merah atas imej tersebut.

1.4 OBJEKTIF

Objektif bagi projek ini adalah:

- a) Untuk mengenalpasti dan menyenaraikan koordinat bagi kawasan yang lebih tumpat dalam kumpulan orang ramai tersebut.
- b) Mengubahsuaikan cara pengiraan jarak antara 2 titik dengan mengimpikasi nilai perspektif.

1.5 SKOP

Projek ini akan mengenalkan kawasan yang lebih tumpat dalam kumpulan orang ramai. Dataset yang akan digunakan dalam projek ini adalah dipanggil membuat ShanghaiTech Dataset oleh Zhang Y Zhou D, Chen S, et al. Dataset ini mengandungi hampir 1,200 imej dengan kepala tepat berlabel yang berjumlah kira-kira 330,000. Dan jenis orang ramai dikenalpasti oleh mengira ruang yang diduduki oleh seseorang dari imej tersebut.

1.6 KEKANGAN

Kekangan untuk projek ini adalah:

- i. Keperluan penyediaan persekitaran untuk menjalankan kod adalah tinggi sedikit untuk persembahan grafik, GTX960.
- ii. Program yang diimplikasikan adalah sukar untuk menyambung antara satu sama lain apabila kod tersebut dijalankan pada program/platform yang berbeza.
- iii. Imej orang ramai mesti adalah satu imej berdefinisi tinggi supaya mengesan jenis orang ramai.

1.7 METODOLOGI

Bagi permulaan projek, alam sekitar akan disediakan termasuk Anaconda 3, python3, Windows 10 dan Spyder 3 IDE. Dataset telah dikenalpasti iaitu ShanghaiTech Dataset dan data nilai perspektif daripada output PACNN.

Kawasan kumpulan orang ramai yang lebih tumpat akan dikesan melalui pengiraan ketumpatan orang ramai dengan DPC (*Density Peak Clustering*).

Output projek ini ialah ketumpatan maxima orang ramai daripada kumpulan orang ramai dan senarai koordinat kawasan orang ramai “mosh-pit” tumpat dan imej orang ramai ditandakan untuk kawasan “mosh-pit” tumpat akan dihasilkan. Semua ini akan dibincangkan dalam perbincangan projek ini.

1.8 JADUAL

Minggu	Decription
1	Mengenalpasti tajuk projek
2	Membuat kajian yang berkaitan journals and maklumat
3	Membuat kajian yang berkaitan journals and maklumat
4	Penyediaan persekitaran kod, D1
5	Penyediaan persekitaran kod
6	Penyediaan persekitaran kod, D2
7	Penyediaan persekitaran kod
8	Penyediaan persekitaran kod
Cuti Pertengahan Semester	
9	Penyediaan persekitaran kod
10	Penyediaan persekitaran kod, D3
11	Penyediaan persekitaran kod
12	Penyediaan persekitaran kod, D4
13	Presentation
14	Pra-KID

Jadual 1.1: Jadual untuk Semester 1

Minggu	Decription
1	Bincang 1
2	Persediaan untuk Demo I
3	Demo I: Pemasangan Peralatan Pembangunan / Pemprosesan
4	Persediaan untuk Demo II
5	Demo II: Pembangunan / Pemprosesan
6	Persediaan untuk Demo III
7	Demo III: Pembangunan / Pemprosesan
Cuti Pertengahan Semester	
8	Penhantaran D5
9	Bincang 2 (D6): Perancangan Pengujian / Penilaian
10	Penhantaran D6
11	Demo IV: Perlaksanaan Pengujian / Penilaian
12	Bincang 3 (D7): Persediaan KID28 Mei 2019 (Selasa): KID
13	Penhantaran D7
14	28 Mei 2019 (Selasa): KID

Jadual 1.2: Jadual untuk Semester 2

1.8 KESIMPULAN

Menurut Kimballyoung "orang ramai adalah majlis perjumpaan besar bilangan orang di seluruh pusat atau titik perhatian umum". Manakala anggaran orang ramai adalah teknik yang penting untuk mengekalkan keselamatan awam dan melaksanakan pengawalan orang ramai. Projek ini akan membantu dalam kefahaman kumpulan orang ramai dengan mengenalpasti jenis orang ramai tersebut. Jenis "mosh-pit" tumpat orang ramai merupakan jenis orang ramai yang mudah berlaku rempuhan manusia. Jenis "mosh-pit" tumpat orang ramai selalu dan berlaku rempuhan manusia menyebabkan banyak kematian dan kecederaan. Kemalangan rempuhan manusia mungkin dapat dihalang apabila orang ramai dikesan terlalu ramai di suatu tempat.

BAB II

KAJIAN KESUSASTERAAN

2.1 PENGENALAN

Kajian kesusastaraan adalah kajian yang ditulis selepas membaca dan mengkaji jurnal-jurnal dan kajian yang berkaitan dengan projek saya. Kajian kesusastaraan yang berkaitan dengan anggaran orang ramai akan dibincangkan dalam bab ini. Dalam bab ini, penerangan tertumpu atas algoritma yang sedia ada dengan algoritma yang saya cadangkan akan dibincangkan.

2.2 LATAR BELAKANG KAJIAN

Rempuhan manusia berlaku pada setiap tahun. (*Mike Rothschild. 2015*) Apabila di suatu tempat yang tutup tetapi terlalu ramai orang, ini akan menyebabkan banyak kematian dan kecederaan apabila berlaku rempuhan manusia. Sekurang-kurangnya 720 Jemaah Islam mati dalam rempuhan semasa upacara mengerjakan Haji pada hari akhir 2015 bulan September. Ia berlaku di sebuah bandar khemah yang beberapa batu dari Mekah. Selain itu, ratusan ribu Hindu bertemu di kuil Mandher Devi, India pada Januari 2005. Silinder gas di sebuah gerai yang terletak di jalan yang sempit telah berlaku peletupan. Ini telah menyebabkan panik kepada orang di situ. Kegagalan pengawalan orang ramai pada situasi situ telah menyebabkan seramai 291 terbunuh dan ratusan orang telah cedera. Semua kemalangan ini mungkin dapat dihalang apabila orang ramai dikesan terlalu tumpat di sesuatu tempat. Oleh itu, kajian tentang pengawalan orang ramai secara automatik telah dimulakan, terutamanya, anggaran orang ramai. Selain itu, rempuhan manusia sentiasa berlaku apabila jenis orang ramai adalah jenis orang ramai yang aktif (*Mob crowd*). Oleh itu, pengesanan jenis orang ramai merupakan jenis yang aktif adalah sangat penting dalam kajian ini.

Dalam bab ini, beberapa kajian yang berkaitan tentang anggaran orang ramai dan jenis orang ramai akan dibincangkan.

2.3 GAMBARAN KESELURUHAN SISTEM

Proses keseluruhan sistem:

1. Pra pemrosesan
2. Pengambilan data nilai perspektif dan nilai GT(*Ground-Truth*)
3. Pengkelompok puncak ketumpatan (*Density Peak Clustering*)

2.3.1 PRA PEMROSESAN

Dalam pra pemrosesan, dataset yang akan digunakan telah dipilih yaitu ShanghaiTech Dataset. Dataset ini mencakup sebanyak 1,200 imej yang mempunyai 330,000 kepala berlabel yang tepat. Persekitaran untuk menjalani kajian akan disediakan seperti penyediaan Python3, Spyder 3 IDE dan Anaconda 3.

2.3.2 PENGAMBILAN DATA NILAI PERSPEKTIF DAN NILAI GT(*GROUND-TRUTH*)

Dalam sektor ini, nilai perspektif dari output PACNN adalah digunakan untuk mengira ketumpatan kumpulan orang ramai. Nilai GT akan didapati daripada dataset ShanghaiTech. Nilai GT ini adalah digunakan sebagai data yang diperlukan untuk sektor yang seterusnya.

2.3.3 PENGKELOMPOK PUNCAK KETUMPATAN (*DENSITY PEAK CLUSTERING*)

Proses ini akan menggunakan nilai GT dengan tarapkan nilai perspektif dalam algoritma pengkelompok puncak ketumpatan supaya mengira ketumpatan kawasan tersebut dan jarak minimum. Proses ini akan memberi beberapa imej orang ramai yang baru untuk mengenalpasti algoritma yang dilaksanakan dapat memberi anggaran orang ramai dan juga mengesan kawasan yang lebih tumpat dalam kumpulan orang ramai tersebut. Set outputnya adalah nombor yang menyatakan anggaran orang ramai, imej yang mempunyai tanda dengan kawasan yang lebih tumpat.

2.4 KAEDAH YANG PERNAH DIGUNAKAN UNTUK ANGGARAN ORANG RAMAI

2.4.1 KAEDAH BERASASKAN PENGESANAN (*DETECTION-BASED METHOD*)

Pengesanan sisi adalah kaedah pengesanan berasaskan satu poplar. Kaedah ini mengeluarkan latar belakang imej oleh rujukan imej rujukan. Perubahan keadaan pencahayaan dapat mengesan bahagian intensiti yang lebih tinggi. Pengesanan sisi adalah kaedah tahap pemprosesan yang kurang baik kerana ketepatannya imej menjatuhkan apabila mengesan orang ramai tersebut tersesak. (*Anthony C. Davies, Jia Hong Yin, Sergio A. Velastin 1995*)

2.4.2 TRAJEKTORI KELOMPOK (*CLUSTER TRAJECTORIES*)

Kaedah ini biasanya digunakan dalam dalam pengesanan berasaskan video. Mereka menggunakan pengesan KLT untuk mengesan pergerakan orang ramai untuk membuat anggaran orang ramai yang bergerak. Pengesan KLT ini juga mengetahui untuk kesederhanaan dan keberkesanan radikal. Kumpulan orang-orang yang bergerak diwakili oleh kelompok objek yang bergerak. Walau bagaimanapun, kaedah ini sesuai dalam video tetapi bukan imej. (*V. Rabaud and Serge Belongie 2006*)

2.4.3 REGRESI BERASASKAN CIRI-CIRI (*FEATURE-BASED REGRESSION*)

Kaedah ini sentiasa digunakan untuk anggaran orang ramai. Kaedah ini perlu membuat segmen terhadap latar hadapan. Cari pelbagai ciri – ciri daripada latar hadapan. Kemudian, menggunakan fungsi regresi untuk menganggarkan bilangan orang ramai. Fungsi regresi digunakan kerana guna fungsi linear akan menurunkan kecekapan. (*A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos 2008*)

2.4.4 RANGKAIAN NEURAL MULTISKALA (*MULTISCALE NEURAL NETWORK*)

Segmen untuk latar hadapan adalah sukar untuk berbuat demikian pada hakikatnya apabila terdapat orang ramai tersesak yang merangkumi keseluruhan garisan. Manakala segmen tidak diperlukan dalam kaedah ini. Kaedah ini menggunakan skala penapis yang berbeza untuk mengesan dalam kiraan kepala dengan menggunakan konsep pengumpulan max. Ia meliputi seluruh bahagian gambar. Saiz penapis yang berbeza skala adalah untuk mengesan berbeza saiz kepala orang ramai. (*Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, Yi Ma 2016*), (*Lingke Zeng, Xiangmin Xu, Bolun Cai, Suo Qiu, Tong Zhang 2017*)

2.4.5 RANGKAIAN NEURAL KONVOLUSI KESEDARAN PERSPEKTIF (*PERSPECTIVE AWARE CONVOLUTIONAL NEURAL NETWORK*)

Dalam cadangan mereka, mereka memperkenalkan peta perspektif kebenaran tanah bagi latihan rangkaian. Ia adalah sebagai fungsi linear paksi tegak bagi satah imej dengan menggunakan geometri projekatif. Fungsi linear ini berdasarkan nilai perspektif yang dikira pada sampel piksel. Mereka mendapati peta ketumpatan dari dua skala yang berbeza daripada rangkaian tulang-belakang tunggal. Rangkaian tersebut mempunyai penapis kecil (cth: $3 * 3$) sahaja dan kemudian menggabungkan dua output ketumpatan melalui lapisan berwajaran sedar perspektif, yang mana pemberat belajar melalui yang peta perspektif ramal yang pengubahan tak linear dalam rangkaian. Output akhir adalah kukuh dengan perspektif yang berbeza-beza dan skala. Mereka mengganggarkan jumlah orang ramai dan ketumpatan orang ramai secara terus melalui imej keseluruhan. (*Miaojing Shi, Zhaohui Yang, Chao Xu, Qijun Chen 2018*)

2.4.6 PERBANDINGAN KAEDAH YANG SEDIA ADA

Perbandingan Kaedah Sedia Ada Kaedah	Pengenalan ketumpatan orang ramai	Kerumitan Proses	Kesesuaian untuk imej	Ketepatan
Kaedah Berasaskan Pengesanan	Tidak	Rendah	Tidak	Rendah
Trajectories Kelompok	Ya	Rendah	Tidak	Tinggi
Regresi Berasaskan Ciri-Ciri	Ya	Tinggi	Ya	Tinggi
Rangkaian Neuron Multiskala	Ya	Rendah	Ya	Tinggi
Rangkaian Neural Konvolusi Kesedaran Perspektif	Ya	Rendah	Ya	Tinggi

Jadual 2.4.6: Perbandingan algoritma sedia ada

2.5 PERKELOMPOKAN

Pada dasarnya, ia adalah sejenis kaedah pembelajaran tanpa penyeliaan. Kaedah pembelajaran tanpa penyeliaan merupakan satu kaedah di mana kita menggunakan rujukan daripada dataset yang terdiri daripada data input tanpa berlabel. Secara umumnya, ia digunakan sebagai satu proses untuk mencari struktur bermakna, penjelasan proses asas, ciri-ciri generatif dan kumpulan sedia ada dalam satu set contoh.

Perkelompokan merupakan tugas membahagikan populasi atau data ke dalam beberapa kumpulan. Populasi dalam kumpulan yang sama lebih menyerupai populasi lain dalam kumpulan yang sama tetapi jauh berbeza dengan populasi dalam kumpulan-kumpulan lain. Pada dasarnya, ia adalah himpunan objek berdasarkan kesamaan dan perbezaan antara mereka.

2.5.1 PERKELOMPOKAN K-MEANS

Matlamat algoritma ini adalah untuk mencari kumpulan dalam data, dengan bilangan kumpulan yang diwakili oleh variabel K. Algoritma ini dijalankan secara berulang untuk memperuntukkan setiap titik data kepada salah satu kumpulan daripada kumpulan K berdasarkan ciri-ciri yang disediakan. Capaian data terkumpul berdasarkan ciri-ciri persamaan. Hasil daripada algoritma perkelompokan K-Means:

- a. Pusat daripada kelompok K, boleh digunakan untuk malabelkan data baru
- b. Label untuk latihan data (setiap titik data diperuntukkan kepada satu kelompok)

Perkelompokan membolehkan anda untuk mencari dan menganalisis kumpulan yang berbentuk organik berbanding dengan memberi label tanpa melihat datanya. Setiap pusat kelompok ialah sekumpulan nilai-nilai ciri-ciri yang menentukan golongan yang terhasil. Perkelompokan K-means ini biasanya digunakan dalam kes-kes perniagaan biasa. Selain itu, K-means juga digunakan sebagai langkah-langkah yang terlibat dalam menjalankan algoritma. (*Andrea Trevino 2016*)

2.5.2 PERKELOMPOKAN DBSCAN

Pengkelompokan spatial berdasarkan kepadatan aplikasi dengan hingar (*Density-based spatial clustering of applications with noise, DBSCAN*) adalah algoritma perkelompokan data terkenal yang biasanya digunakan dalam perlombongan data dan mesin pembelajaran. Berdasarkan satu set mata, DBSCAN mengumpulkan mata yang dekat bersama-sama antara satu sama lain berdasarkan ukuran jarak (biasanya jarak Euclidean) dan untuk meminimumkan bilangan mata. Ia juga menandakan mata yang berada di dalam kawasan-kawasan yang berkepadatan sebagai outliers. Algoritma DBSCAN memerlukan 2 parameter:

- a. EPS: jarak minimum di antara dua titik. Ini bermakna bahawa jika jarak di antara dua titik adalah lebih rendah atau sama dengan nilai ini (eps), mata ini dianggap jiran-jiran.
- b. minPoints: bilangan minimum mata untuk membentuk suatu kawasan yang tumpat. Sebagai contoh, jika kita setkan parameter minPoints sebagai 5, maka kita perlu sekurang-kurangnya 5 mata untuk membentuk suatu kawasan yang tumpat.

Anggaran parameter merupakan masalah untuk setiap tugas perlombongan data. Untuk memilih parameter yang baik kita perlu memahami bagaimana mereka akan menggunakan dan mempunyai sekurang-kurangnya sedikit pengetahuan terdahulu mengenai set data yang akan digunakan.

- a. EPS: jika nilai eps yang dipilih terlalu kecil, sebahagian besar daripada data tidak akan berkelompok. Ia akan dikira sebagai outlier kerana tidak memenuhi bilangan mata untuk mencipta suatu kawasan tumpat. Sebaliknya, jika nilai yang telah dipilih adalah terlalu tinggi, kelompok akan bergabung dan majoriti objek akan dalam kumpulan yang sama. Eps patut dipilih berdasarkan pada jarak dataset, tetapi secara umum nilai eps yang kecil adalah lebih baik.
- b. minPoints: minPoints minimal yang boleh diperolehi daripada beberapa dimensi (D) dalam data set, sebagai $\text{minPoints} \geq D + 1$. Nilai-nilai yang lebih besar biasanya lebih baik bagi set data dengan hingar dan akan menjadi kelompok yang utama. Nilai minimum untuk minPoints itu

mestilah 3, tetapi set data yang lebih besar, nilai minPoints yang lebih besar patut dipilih.

Algoritma DBSCAN yang perlu digunakan untuk mencari kelompok dan struktur dalam data yang sukar untuk mencari secara manual tetapi ia boleh dianggap relevan dan berguna untuk mencari pola-pola dan meramalkan aliran. Kaedah penggugusan biasanya digunakan dalam biologi, perubatan, sains sosial, arkeologi, pemasaran, pengecaman aksara, sistem pengurusan dan sebagainya. (*Kelvin Salton do Prado 2017*)

2.5.3 PERKELOMPOKAN PUNCAK KETUMPATAN

Sama dengan kaedah K-medoids, algoritma ini mempunyai jarak asas di antara titik data. Seperti DBSCAN dan kaedah min-shift, ianya mampu untuk mengesan kelompok yang bukan bentuk sfera dan mencari secara automatik bilangan kelompok yang betul. Pusat kelompok adalah ditakrifkan, seperti dalam kaedah min-shift, sebagai maxima tempatan dalam ketumpatan capaian data. Walau bagaimanapun, tidak seperti kaedah min-shift, prosedur kami tidak memerlukan benaman data dalam ruang vektor dan memaksimumkan jelas medan ketumpatan bagi setiap titik data. Algoritma yang mempunyai asas dengan andaian bahawa kelompok pusat dikelilingi oleh jiran-jiran dengan lebih rendah ketumpatan tempatan dan bahawa mereka berada di dalam jarak yang agak besar dari sebarang pusat kelompok dengan ketumpatan tinggi tempatan. (*Miaojing Shi, Zhaohui Yang, Chao Xu, Qijun Chen 2018*)

2.6 KESIMPULAN

Dalam projek ini, kami akan mendapatkan nilai perspektif dari PACNN dan nilai kebenaran tanah kumpulan orang ramai daripada dataset ShanghaiTech. Kefahaman kumpulan orang ramai adalah sangat penting kerana rempuhan manusia berlaku di kawasan yang sangat tumpat. Oleh yang demikian, projek ini akan pengesanan kawasan yang lebih tumpat di kumpulan orang ramai. Pengiraan adalah berdasarkan ruang yang dihuni oleh setiap orang dengan menggunakan nilai perspektif dan nilai kebenaran tanah dengan perkelompokan puncak ketumpatan. Output bahagian ini akan menjadi satu imej orang ramai yang ditandakan kawasan yang lebih tumpat.

Kesimpulannya, algoritma ini akan membantu untuk meningkatkan keselamatan isu dengan mengelakkan rempuhan manusia yang berlaku.

BAB III

SYARAT – SYARAT SPESIFIKASI

3.1 PENGENALAN

Anggaran kumpulan orang ramai dan kefahaman kumpulan orang ramai adalah teknik yang penting untuk mengekalkan keselamatan awam dan melaksanakan kawalan kumpulan. Apabila tempat terlalu sesak, rempuhan manusia akan berlaku dan menyebabkan kematian dan luka yang tidak terkira banyaknya. Kefahaman kumpulan orang ramai adalah berguna untuk mengelakkan kemalangan ini berlaku kerana tempat yang terlalu ramai akan dapat dikesan.

Pada masa ini anggaran sesak telah dilakukan penyelidikan oleh beberapa pasukan dengan cara yang berbeza. Anggaran berasaskan pengesanan kumpulan dilakukan oleh P. Viola, M. J. Jones dan D. Snow (*P. Viola, M. J. Jones, and D. Snow 2005*). Kaedah ini telah berjaya mengesan jumlah kiraan kepala di kalangan orang ramai, tetapi prestasi pengesanan akan terjejas apabila terdapat kumpulan yang sesak maka ia turut menjejaskan ketepatan. Adapun regresi berasaskan ciri-ciri, ia adalah lebih kompleks untuk diguna pakai dalam anggaran kumpulan kerana ia perlu cabutan dalam segmen latar hadapan supaya meneruskan regresi fungsi anggaran kumpulan. Suatu algoritma yang dicadangkan oleh satu lagi kumpulan penyelidik dari Universiti ShanghaiTech yang iaitu rangkaian neural pelbagai skala. (*MA Security Group 2017*) Kaedah ini mempunyai ketepatan yang tinggi apabila mengesan pengiraan kepala dalam kumpulan kerana mereka perlu menggunakan saiz penapis yang berbeza untuk berdepan dengan skala yang berbeza daripada kepala. Namun begitu orang adalah dalam pelbagai perspektif, untuk menghadapi kepala skala berbeza, algoritma ini adalah agak kompleks kerana pelbagai column/rangkaian memperkenalkan lebih banyak parameter untuk mengambil sumber pengkomputeran yang lebih banyak, ini menyukarkan untuk praktikal permohonan. Demi mendapatkan nilai perspektif, dataset PACNN telah digunakan. Demi mendapatkan nilai kebenaran tanah, dataset ShanghaiTech telah digunakan. Selain itu, demi mendapatkan jarak antara 2 orang, maka perkelombongan puncak ketumpatan telah diimplikasi dalam projek ini. Menurut

ke Hebert Jacobsen, jika ruang yang diduduki seseorang adalah kurang daripada 0.50 meter persegi ruang, maka ia adalah kawasan “mosh-pit” tumpat. Kawasan ini akan ditandakan pada imej asal. Ini dilakukan supaya kawasan ini harus dielakkan dengan serta-merta agar mengelakkan rempuhan manusia berlaku.

3.2 DEFINISI KEPERLUAN SISTEM

Projek ini dicadangkan untuk GPU daripada GTX960 dan ke atas. Dalam projek ini, persediaan persekitaran adalah Python3, Spder 3 IDE dan Anaconda 3 untuk menjalankan kod. Untuk persediaan persekitaran ini, Linux 14.04 adalah disyorkan. Dataset yang digunakan adalah ShanghaiTech Dataset yang berasal dari Universiti ShanghaiTech. Dataset ini adalah lebih boleh dipercayai kerana ia mengandungi hampir 1,200 imej dengan kepala tepat dilabelkan berjumlah kira-kira 330,000. Ia mempunyai kumpulan terbesar mengira dataset dari segi kepala annotated nombor. Tiada kedua-dua imej dalam dataset ini diambil daripada sudut pandangan sama. Dataset ini terdiri daripada dua bahagian: Bahagian A dan Bahagian B. imej dalam Bahagian A secara rawak merangkak dari Internet, kebanyakan mereka mempunyai sebilangan besar rakyat. Bahagian B diambil daripada jalan-jalan yang sibuk di kawasan metropolitan di Shanghai.

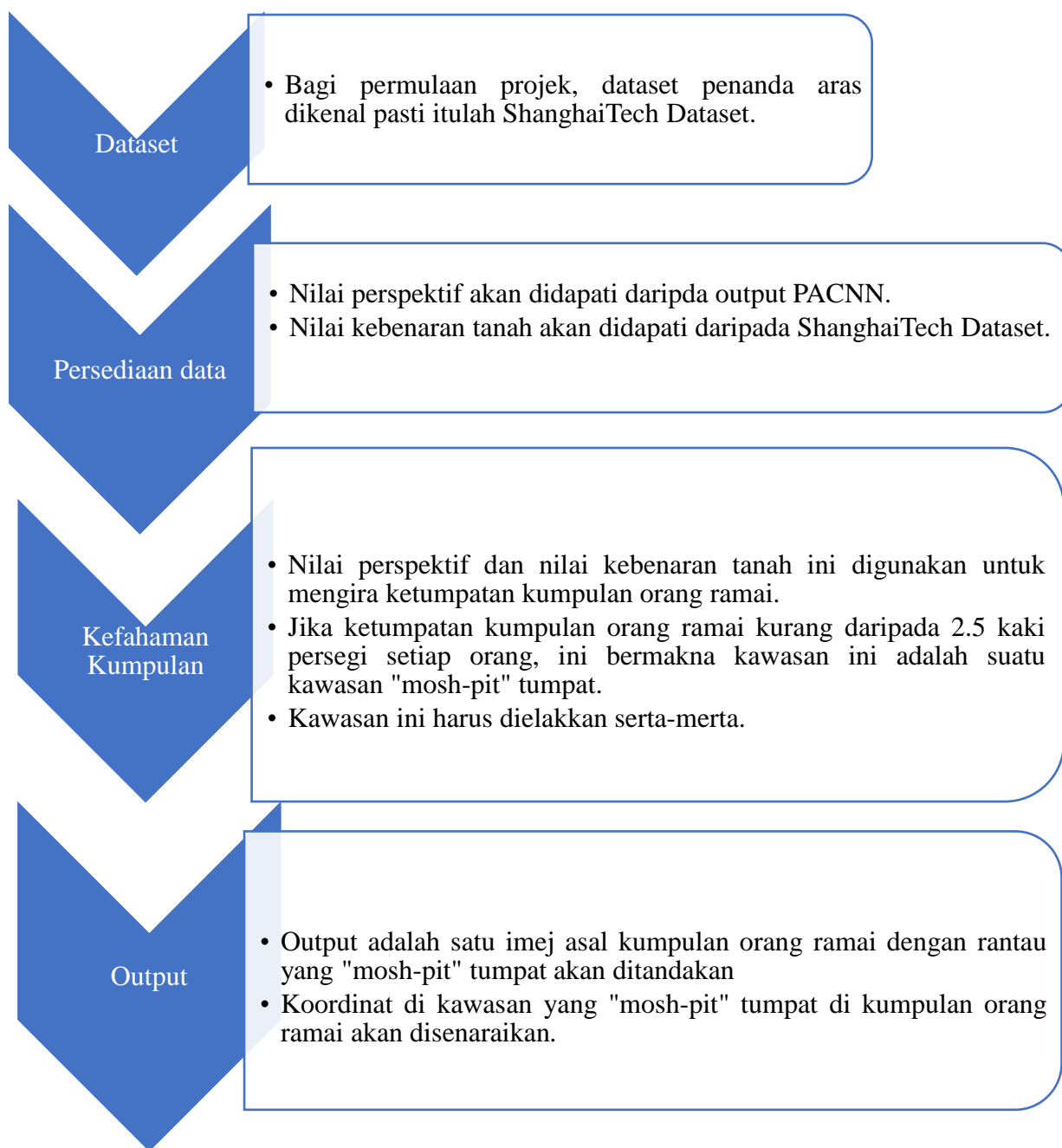
3.3 MODEL SYSTEM

Nilai perspektif dari PACNN dan nilai kebenaran tanah(*Ground-Truth, GT*) daripada dataset ShanghaiTech didapatkan bagi proses pengiraan.

Pengiraan ketumpatan akan dilakukan dengan algoritma perkelompokan puncak ketumpatan. Terdapat empat kategori asas untuk ketumpatan orang ramai, yang kurang tumpat, sederhana tumpat lebih padat dan kumpulan orang ramai yang “mosh-pit” tumpat. Ini adalah berdasarkan ruang yang dihuni oleh seseorang di dalam orang ramai, jika seseorang yang menghuni lebih 1.5 meter persegi ruang, ini bermaksud ia adalah suatu kawasan yang kurang tumpat. Manakala jika seseorang menghuni lebih daripada 1 meter persegi ruang tetapi kurang 1.5 meter persegi ruang, ia bermakna ia ialah kumpulan orang ramai yang sederhana padat. Jika seseorang menghuni lebih daripada 0.5 meter persegi ruang tetapi kurang 1.0 meter persegi ruang, ia bermakna ia ialah kumpulan orang ramai yang lebih padat. Jika ruang yang dihuni oleh seseorang adalah kurang daripada 0.5 meter persegi ruang, maka ia adalah suatu kawasan “mosh-pit” tumpat. Ini adalah kawasan berbahaya yang rempuhan manusia mungkin berlaku di kawasan ini pada bila-bila masa.

Jikalau kawasan itu dikira sebagai kawasan “mosh-pit” tumpat, maka dalam imej kumpulan orang ramai, kawasan tersebut akan ditandakan dengan titik merah.

Output bagi projek ini adalah imej kumpulan orang ramai dengan ditandakan kawasan “mosh-pit” tumpat dan senarai koordinat untuk kawasan yang “mosh-pit” tumpat.



Rajah 3.3.1: Seni bina projek

3.4 KESIMPULAN

Dalam projek ini, algoritma perkelompokan puncak ketumpatan digunakan untuk pengiraan jarak antara 2 orang. Output adalah imej yang ditanda titik merah jika ada kawasan “mosh-pit” tumpat atau kawasan yang paling tumpat dalam imej tersebut. Jenis kumpulan yang akan dikesan dengan pengiraan jarak antara orang kumpulan. Projek ini adalah untuk membantu dalam pencegahan rempuhan manusia kerana ia boleh mengesan jenis kumpulan itu sama ada adalah kumpulan ‘mosh-pit’ yang akan menyebabkan rempuhan manusia. Kiraan kepala adalah penting untuk mengetahui jika tempat yang terlalu ramai menyebabkan rempuhan manusia juga.

Untuk menjalankan projek ini, ia adalah perlu mempunyai sebuah GPU GTX960 dan ke atas. Keperluan perisian dan pakej untuk memasang akan Python3, Spyder 3 IDE and Anaconda 3 and Linux 14.04 untuk menjalankan kod.

Selepas persediaan alam sekitar bagi projek tersebut, keseluruhan projek boleh dijalankan dengan melaksanakan kod kefahaman kumpulan orang ramai. Nilai perspektif dan nilai kebenaran tanah akan didapatkan dari PACNN dan ShanghaiTech dataset akan digunakan untuk mengira ketumpatan kumpulan orang ramai. Jika ketumpatan kumpulan orang ramai kurang daripada 0.5 meter persegi setiap orang, ini bermakna kawasan ini adalah suatu kawasan yang “mosh-pit” tumpat. Kawasan ini harus dielakkan dengan serta-merta.

BAB IV

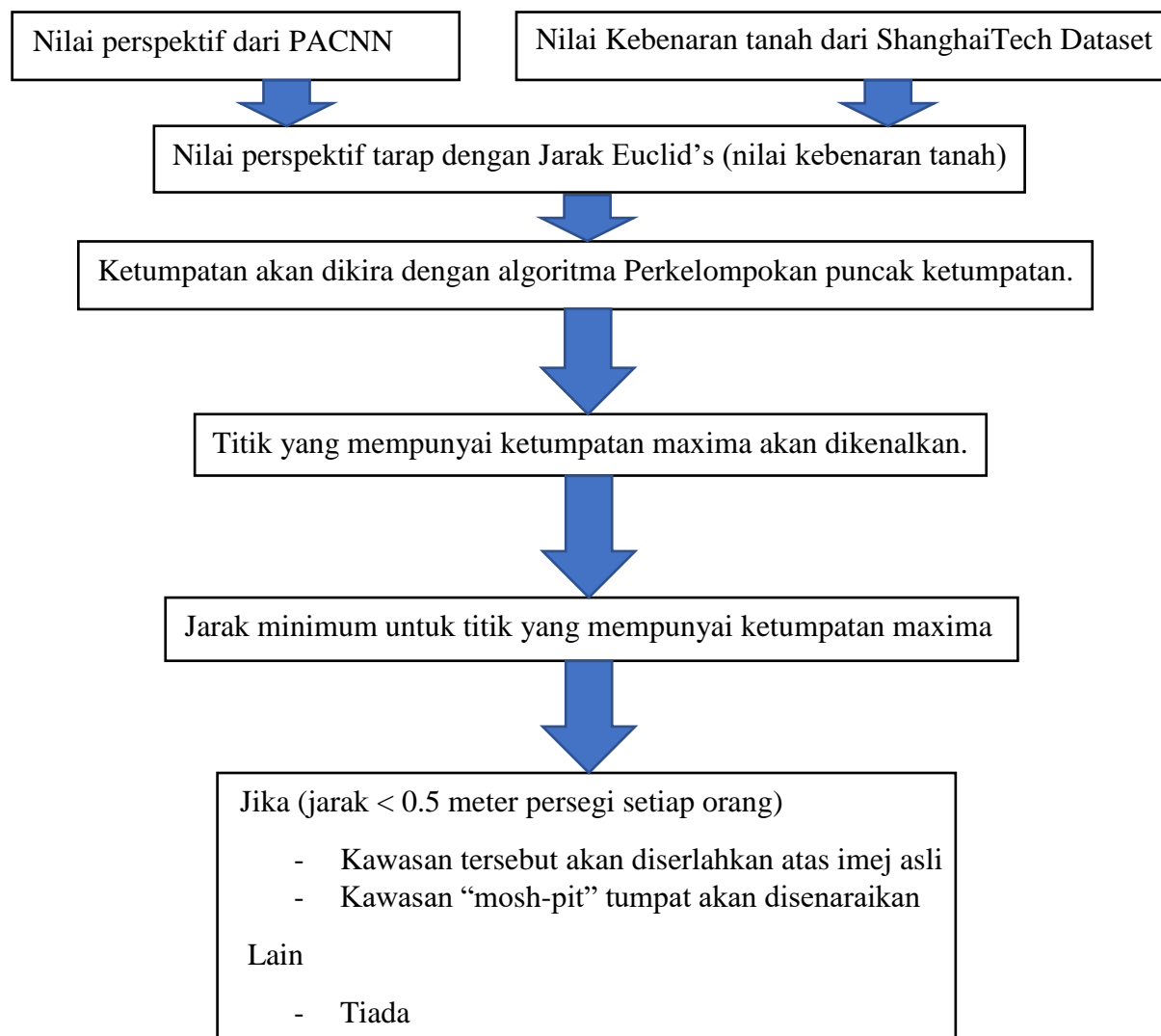
SPESIFIKASI REKA BENTUK

4.1 PENGENALAN

Dalam bab ini, reka bentuk seni bina dan reka bentuk algorima akan dibincangkan. Selepas bab ini, seluruh algorima akan dibincangkan dan proses projek ini akan dibincangkan.

Yang pertama sekali adalah mendapat nilai perspektif and nilai *Ground Truth* (GT) dengan daripada rangkaian neural konvolusi kesedaran perspektif, *Perspective Aware Convolutional Neural Network* (PACNN) dan ShnghaiTech dataset. (*Miaojing Shi, Zhaohui Yang, Chao Xu, Qijun Chen 2018*) Selepas mendapat bahan-bahan yang diperlukan, pengiraan ketumpatan akan dilakukan dengan mengimplikasikan algoritma perkelompokan puncak ketumpatan. Untuk mengetahui jika kawasan itu merupakan kawasan yang lebih tumpat, kita perlu menggunakan nilai perspektif tarap dengan pengiraan jarak antara titik kepala supaya mendapatkan ruang sebenar yang diduduki oleh seseorang. Jarak minimum akan digunakan untuk mengategorikan kumpulan tersebut. Jikalau jarak tu kurang daripda 0.5 meter persegi, maka kawasan itu merupakan kawasan yang “mosh-pit” dan kawasan inilah yang mudah berlaku rempuhan manusia pada bila – bila masa. Kawasan ini akan diserlahkan dalam imej asli dan koordinat kawasan tersebut akan disenaraikan. Kawasan ini dikesan untuk meningkatkan kesedaran orang ramai tentang risiko berlaku rempuhan manusia dan rantau ini harus mengelak daripada serta-merta.

4.2 REKA BENTUK SENI BINA



Rajah 4.2.1: Reka bentuk seni bina

Nilai perspektif yang diperolehi daripada PACNN dan nilai kebenaran tanah yang diperolehi daripada ShanghaiTech dataset adalah bersedia untuk mencari jarak antara satu titik dengan lain. Algoritma Perkelompokan puncak ketumpatan akan diubahkan dengan tarapkan nilai perspektif dalam jarak Euclid's. Manakala, nilai kebenaran tanah telah bersedia dalam ShanghaiTech dataset akan diekstrak pada permulaan projek ini. Nilai perspektif digunakan untuk mengira nilai sebenar jarak imej. Ini adalah penting bagi proses pengiraan seterusnya.

Setelah mendapat jarak sebenar, pengiraan ketumpatan akan dilakukan dengan algoritma perkelompokan puncak ketumpatan:

$$\rho_i = \sum_j X(d_{ij} - d_c) \times \text{nilai perspektif}$$

Selepas itu, ketumpatan maxima akan dikira dengan mengambil

$$\delta_i = \max_j(d_{ij})$$

Daripada

$$\delta_i = \min_{j:\rho_j > \rho_i}(d_{ij})$$

Jarak minimum akan diambil telah mencari ketumpatan yang paling tinggi. Selepas pengiraan ketumpatan maxima jarak minimum antara titik tersebut dengan titik jirannya, jarak tersebut akan diambil dan dibandingkan. Jika ruang antara setiap orang adalah kurang daripada 0.5 meter persegi, bermaksud ia adalah kawasan kumpulan orang ramai “mosh-pit” tumpat. Manakala, jika ruang antara setiap orang adalah lebih daripada 0.5 kaki persegi dan kurang 1.0 meter persegi adalah kawasan yang lebih tumpat. jika ruang antara setiap orang adalah lebih daripada 1.0 kaki persegi dan kurang 1.5 meter persegi adalah kawasan yang sederhana tumpat. Selebihnya adalah kawasan yang kurang padat dalam imej. Koordinat di kawasan ini sangat tumpat direkodkan dan kawasan sangat tumpat ini akan diserlahkan dalam imej asli.

4.3 REKA BENTUK ALGORITMA

Kod pseudo:

```
def distanceNorm(Norm,D_value):{
    Norm untuk Jarak:
    Jika Norm =1:
        counter = Jarak;
        counter += counter;
    atau jika Norm =2:
        counter = Jarak2;
        counter+ = counter;
        counter =  $\sqrt{\text{counter}}$ ;
    atau jika Norm =  $\infty$ :
        counter = jarak;
        counter = Max(counter);
    atau:
        penghalang pengecualian: ('We will program this
later.....');
        kembalikan counter;
}
def chi(x):{
    jika x < 0:
        kembalikan 1;
    atau:
        kembalikan 0;
}
def fit(ciri-ciri,label,t,distanceMethod = '2'){
    Penamaan
    dapatkan nilai perspektif dari PACNN
    pengiraan jarak dengan menarapkan
    Bagi i dalam jangka(len(labels)):
        Bagi j dalam jangka(i+1,len(labels)):
            Dapatkan lokasi[i] dan lokasi[j] untuk pengiraan
            jarak;
            Dapatkan nilai perspektif untuk lokasi[i] dan
            lokasi[j];
            (Nilai perspektif [i]+ Nilai perspektif [j] )/2
```

```

        Dapatkan jarak antara dua titik kepala dengan
        algoritma Euclid's dan tarap dengan nilai perspektif
    Pengiraan cutoff
    Pengiraan ketumpatan:
    Bagi i dalam jangka(len(labels)):
        Cutoff = distance[i]- cutoff;
        Bagi j dalam jangka(1,len(labels)):
            Ketumpatan[i] += chi(cutoff[j]);
    Mencari ketumpatan maxima;
    Dapatkan koordinat titik yang mempunyai ketumpatan maxima;
    Dapatkan jarak minimal untuk titik tersebut;
    Jika jarak tersebut < 0.5
        Cetak ("kawasan "mosh-pit" tumpat");
    Atau jika : 0.5<jarak tersebut<1.0:
        Cetak("kawasan yang sangat tumpat");
    Atau jika : 1.0<jarak tersebut<1.5:
        Cetak("kawasan yang sederhana tumpat");
    Lain daripada itu:
        Cetak("kawasan kurang tumpat");
    Kembalikan (ketumpatan);
}

input pengguna
dapatkan nilai kebenaran tanah imej mengikuti input pengguna
plot untuk koordinat titik kepala setiap orang dalam imej
tersebut
penggiraan ketumpatan dan jarak minimum dengan fit();
plot untuk koordinat kawasan "mosh-pit"

```


4.4 KESIMPULAN

Kesimpulannya, reka bentuk seni bina adalah berdasarkan perkolompokan puncak ketumpatan (*Density peak Clustering*) untuk mendapatkan ketumpatan. Pada pertama, nilai perspektif akan didapati dari output PACNN dan nilai kebenaran tanah akan didapati daripada dataset ShanghaiTech. Nilai kebenaran tanah adalah untuk mengira jarak antara titik kepala dan ketumpatan kawasan tersebut. Manakala nilai perspektif akan ditarapkan dalam perkelompokan puncak ketumpatan apabila pengiraan jarak. Titik yang maxima ketumpatan akan dijejaskan dan minimum jarak antara titik tersebut dengan titik jiran akan direkodkan. Jarak tersebut digunakan untuk berbanding dengan nilai ambang yang berdasarkan jurnal Dillion. Menurut 1960's Herbert Jacobsen, seorang Professor kewartawanan, jika ruang antara setiap orang adalah kurang daripada 0.5 meter persegi, bermaksud ia adalah kawasan kumpulan orang ramai "mosh-pit" tumpat. Manakala, jika ruang antara setiap orang adalah lebih daripada 0.5 kaki persegi ruang dan kurang 1.0 meter persegi ruang adalah kawasan yang lebih tumpat. jika ruang antara setiap orang adalah lebih daripada 1.0 meter persegi ruang dan kurang 1.5 meter persegi ruang adalah kawasan yang sederhana tumpat. Selebihnya adalah kawasan yang kurang padat dalam imej. Rempuhan manusia biasa berlaku pada kawasan "mosh-pit" tumpat dan kawasan ini akan diserlahkan di gambar asli dan koordinat kawasan ini akan disenaraikan.

BAB V

PMBANGUNAN SISTEM DAN PENGUJIAN

5.1 PENGENALAN

Bab ini akan menjelaskan pembangunan sistem dan pengujian yang telah dilakukan selepas bab sebelum ini. Bab ini akan merealisasikan reka bentuk sistem yang dibincangkan daripada bab sebelum ini and pembangunan sistem adalah berdasarkan reka bentuk tersebut. Bab ini akan menjelaskan kod kritikal dengan jelas tentang algoritma yang saya gunakan iaitu algoritma perkelompokan puncak ketumpatan dan cara saya ubahsuaikan and meimplikasikan dalam kod saya. Bab ini akan menunjukkan sebahagian keputusan daripada algoritma saya.

5.2 PEMBANGUNAN SISTEM

5.2.1 SEGMENT KOD KRITIKAL

```
#####-----MAIN_START-----#####
try:
    test = input("Please select which set of images is going to test: \nA. Test A\nB. Train A\nC. Test B\nD. Train B
    while test == 'A' or test == 'B' or test == 'C' or test == 'D':
        if test == 'A':
            string = input("Please insert number 1 to 182 to select image: ")
        elif test == 'B':
            string = input("Please insert number 1 to 300 to select image: ")
        elif test == 'C':
            string = input("Please insert number 1 to 316 to select image: ")
        else:
            string = input("Please insert number 1 to 400 to select image: ")
        print("Please wait a while...")
        # get Ground-Truth value
        GT = GTmatload()
        # density peak clustering
        algo = fit(GT[0], GT[1], 0, distanceMethod = '2')
        # plot second graph
        imgShow()
    test = input("Please select which set of images is going to test: \nA. Test A\nB. Train A\nC. Test B\nD. Tra
except:
    print("exit...")
```

Rajah 5.2.1.1: Kod utama

Kod utama adalah mendapatkan input pengguna dan mencari imej daripada folder yang tertentu. GTmatload(), fit(), imgShow() akan dijalankan. Pengiraan ketumpatan adalah dalam kaedah fit(). Program ini akan ulang sekiranya input pengguna adalah A, B, C, atau D bagi soalan memilih folder.

5.2.2 KOD UNTUK MENAMBAH TUGASAN

Kod perkelombokan puncak ketumpatan(*Density Peak Clustering*) diimplikasi dalam projek ini. Ini disebabkan oleh algoritma ini dapat menganalisis ketumpatan mengikuti data yang diberi. Bahagian seterusnya akan menjelaskan tentang kod algoritma perkelombokan puncak ketumpatan.

```
def distanceNorm(Norm, D_value):
    # initialization
    # Norm for distance
    if Norm == '1':
        counter = np.absolute(D_value)
        counter = np.sum(counter)
    elif Norm == '2':
        counter = np.power(D_value, 2)
        counter = np.sum(counter)
        counter = np.sqrt(counter)
    elif Norm == 'Infinity':
        counter = np.absolute(D_value)
        counter = np.max(counter)
    else:
        raise Exception('We will program this later.....')
    return counter
```

Rajah 5.2.2.1: Kod untuk distanceNorm()

Dalam kod distanceNorm(), ia memngandungi variable *Norm* dengan *D_value*. Norm merupakan norma dalam algoritma manakala *D_value* adalah set koordinat untuk dikira jaraknya. Kod ini adalah untuk mendefinisikan algoritma Euclid's. Dalam projek ini, Norma yang kita guna adalah norma yang kedua. Itu bermakna jarak antara dua titik adalah dikira dengan punca kuasa dua koordinat $[i]-[j]$.

```
def chi(x):
    if x < 0:
        return 1
    else:
        return 0
```

Rajah 5.2.2.2: Kod untuk chi()

Chi() merupakan jalan ringkas untuk algoritma ini. Ia digunakan untuk memisahkan kelompok-kelompok supaya algoritma ini tidak dijalankan menjadi satu kelompok sahaja. Variabel X haruslah ditentu oleh pengguna.

```

File Edit Format Run Options Window Help
def fit(features, labels, t, distanceMethod = '2'):
    # initialization
    # get perspective value
    # perspective value= 8, 8 pixels in one meter, which one pixel equals 1/8
    if test == 'A':
        perspectivematload = loadmat("PACNN-PerspectiveAware-CrowdCounting-master/perspective-ShanghaiTech/A/test_pm
        finaldata = perspectivematload['pmap']
    elif test == 'B':
        with h5py.File("PACNN-PerspectiveAware-CrowdCounting-master/perspective-ShanghaiTech/A/train_pmap/IMG_"+stri
        data = [np.array(element) for element in f['pmap']]
        newdata = np.flip(data,axis=1)
        finaldata = np.rot90(newdata,1,(0,1))
    elif test == 'C':
        with h5py.File("PACNN-PerspectiveAware-CrowdCounting-master/perspective-ShanghaiTech/B/test_pmap/IMG_"+stri
        data = [np.array(element) for element in f['pmap']]
        newdata = np.flip(data,axis=1)
        finaldata = np.rot90(newdata,1,(0,1))
    else:
        with h5py.File("PACNN-PerspectiveAware-CrowdCounting-master/perspective-ShanghaiTech/B/train_pmap/IMG_"+stri
        data = [np.array(element) for element in f['pmap']]
        newdata = np.flip(data,axis=1)
        finaldata = np.rot90(newdata,1,(0,1))
    perspectivedata = pd.DataFrame(finaldata)
    perspectivedata.to_csv('P_B.csv', index=False)
    pdata = pd.read_csv('P_B.csv')
    distance = np.zeros((len(labels), len(labels)))
    distance_sort = list()
    density = np.zeros(len(labels))

```

Rajah 5.2.2.3: Kod untuk fit() – Bahagian 1(a)

```

# compute distance with perspective value
for index_i in range(len(labels)+1):
    for index_j in range(index_i+1, len(labels)):
        pValueOfI = pdata.iat[int(features[index_i, 1]), int(features[index_i, 0])]
        pValueOfJ = pdata.iat[int(features[index_j, 1]), int(features[index_j, 0])]
        avgPValue = (pValueOfI + pValueOfJ)/2
        D_value = features[index_i] - features[index_j]
        distance[index_i, index_j] = distanceNorm(distanceMethod, D_value)/ avgPValue
        distance_sort.append(distance[index_i, index_j])
    distance += distance.T

```

Rajah 5.2.2.4 Kod untuk fit() – Bahagian 1(b)

Bagi algoritma perkelombokan puncak ketumpatan, fit() ini merupakan kod tulang belakang bagi algoritma ini. Yang pertama sekali, variabel features merupakan set koordinat, labels merupakan index bagi set koordinat tersebut, manakala t merupakan nombor yang menentukan “cutoff” bagi chi(), dan distanceMethod = “2”. Langkah yang pertama adalah importkan data nilai perspekif untuk imej tersebut daripada output PACNN. Setelah mendapatkan nilai perspektif, bagi permulaan kaedah ini, variabel yang akan digunakan akan dimulakan. Bagi nilai perspektif yang akan digunakan, akan menurut koordinat titik kepala, maksudnya, titik kepala $[i]$ yang mempunyai nilai perspektif $[p_i]$ akan tambah dengan titik kepala $[j]$ yang mempunyai nilai perspektif $[p_j]$ dan akan dibahagikan dengan 2. Bahagian ini yang saya telah ubahsuaikan kerana nilai perspektif dalam imej akan mengesan dalam pengiraan jarak antara dua titik kepala. Nilai perspektif dapat menapis kedudukan titik kepala dalam imej tersebut adalah lebih ke hadapan berbanding yang lain.

```

File Edit Format Run Options Window Help
# compute optimal cutoff
distance_sort = np.array(distance_sort)
cutoff = int(np.round(distance_sort[len(distance_sort) * t]))
# computer density
for index_i in range(len(labels)):
    distance_cutoff_i = distance[index_i] - cutoff
    for index_j in range(1, len(labels)):
        density[index_i] += chi(distance_cutoff_i[index_j])
# search for the max density
Max = np.max(density)
MaxIndexList = list()
for index_i in range(len(labels)):
    if density[index_i] == Max:
        MaxIndexList.extend([index_i])
        indexofMax = index_i
locationofMax = labels[indexofMax]
print("Max Density =", Max, "\nIndex of Max Density Point =", indexofMax, "\nLocation for Max Density =", locati
plt.plot(locationofMax[0], locationofMax[1], 'ro')
# get smallest distance of the max density point
mindistancethreshold = 10000000000
aaa = distance[indexofMax].tolist()
for i in range(len(aaa)):
    if aaa[i] > 0 and aaa[i] < mindistancethreshold:
        mindistancethreshold = aaa[i]
print("min distance for max density point =", mindistancethreshold)
densityReal = (mindistancethreshold*2)*math.pi
print("Density: ", densityReal)

```

Rajah 5.2.2.5: Kod untuk fit() – Bahagian 2

Jarak antara dua titik kepala akan dikira menggunakan distanceNorm(), algoritma Euclid's. Data jarak akan disimpan dengan bentuk berbaris berikut:

Jarak(I,J)
Jarak(I ₁ , J ₁)
Jarak(I ₁ , J ₂)
Jarak(I ₁ , J ₃)
Jarak(I ₂ , J ₁)
Jarak(I ₂ , J ₂)
Jarak(I ₂ , J ₃)

Jadual 5.2.2.1: Pengiraan jarak

Seterusnya, nilai laluan akan ditarapkan dengan jarak yang telah dikira. Ketumpatan akan dikomputkan. Selepas itu, ketumpatan maxima dan indeks titik tersebut akan direkodkan supaya dapat jejak koordinat titik kepala tersebut.

```
# categorize crowd
if (densityReal < 0.5):
    print("Mosh-pit Crowd")
elif (0.5 <= densityReal < 1.0):
    print("Heavy Dense Crowd")
elif (1.0 <= densityReal < 1.5):
    print("Medium Dense Crowd")
else:
    print("Light Dense Crowd")
return (density)
```

Rajah 5.2.2.6: Kod untuk fit() – Bahagian 3

Selepas dapat koordinat titik kepala yang mempunyai ketumpatan maxima, jarak minimal untuk titik kepala tersebut juga dijejakkan. Jarak ini adalah mewakili jarak yang paling kecil antara titik kepala tersebut dengan titik kepala jirannya. Maksudnya, kita boleh guna jarak minimal ini untuk mengesahkan kawasan tersebut merupakan kawasan “mosh-pit” tumpat atau tidak. Dengan ini, jika jarak minimal ini kurang daripada 0.5 meter persegi, maka kumpulan ini adalah kumpulan “mosh-pit” tumpat. Namun begitu, jarak minimal ini adalah dalam meter, oleh itu, untuk mendapatkan jarak minimal dalam meter persegi, jarak minimal kuasa dua kemudian tarap dengan π .

```
def GTmatload():
    if test == 'A':
        GTmatload = loadmat('ShanghaiTech/ShanghaiTech/part_A/test_data/ground-truth/GT_IMG_'+strimg+'.mat')
    elif test == 'B':
        GTmatload = loadmat('ShanghaiTech/ShanghaiTech/part_A/train_data/ground-truth/GT_IMG_'+strimg+'.mat')
    elif test == 'C':
        GTmatload = loadmat('ShanghaiTech/ShanghaiTech/part_B/test_data/ground-truth/GT_IMG_'+strimg+'.mat')
    else:
        GTmatload = loadmat('ShanghaiTech/ShanghaiTech/part_B/train_data/ground-truth/GT_IMG_'+strimg+'.mat')
    featureofGT = GTmatload["image_info"]
    # convert ndarray to array, export to new csv in array
    a = featureofGT[0, 0]
    b = a[0, 0]
    GTdata = b[0].tolist()
    GTdatapath = pd.DataFrame(data=GTdata)
    GTdatapath.to_csv('GTValue.csv', index=False)
    # Importing the GT dataset
    data = pd.read_csv('GTValue.csv')
    data.head()
    # Getting the values and plotting it
    f1 = data['0'].values
    f2 = data['1'].values
    fitFeature = np.array(list(zip(f1, f2)))
    # plot first graph
    plt.title('Head position in the image')
    plt.gca().invert_yaxis()
    plt.scatter(f1, f2, c='black')
    plt.show()
    print("Please wait a while for calculation")
    return fitFeature, GTdata
```

Rajah 5.2.2.7 Kod untuk Gtmatload()

Kod ini digunakan untuk mendapat imej daripada folder yang tertentu. Selepas itu, nilai perspektif imej tersebut akan load dalam file csv dan digunakan dalam fit().

```
def imgShow():  
    if test == 'A':  
        img = plt.imread('ShanghaiTech/ShanghaiTech/part_A/test_data/images/IMG_'+strimg+'.jpg')  
    elif test == 'B':  
        img = plt.imread('ShanghaiTech/ShanghaiTech/part_A/train_data/images/IMG_'+strimg+'.jpg')  
    elif test == 'C':  
        img = plt.imread('ShanghaiTech/ShanghaiTech/part_B/test_data/images/IMG_'+strimg+'.jpg')  
    else:  
        img = plt.imread('ShanghaiTech/ShanghaiTech/part_B/train_data/images/IMG_'+strimg+'.jpg')  
    plt.title("image with most dense point")  
    plt.imshow(img)  
    plt.show()
```

Rajah 5.2.2.8 Kod untuk imgShow()

Selepas pengiraan ketumpatan, imej asal akan ditimbulkan dengan tanda merah yang menunjukan kawasan yang paling tumpat dalam imej tersebut.

5.3 PELAN PENGUJIAN

Jenis pengujian saya adalah berdasarkan ujian bukan kefungsian (Non – functional Testing) dan teknik ujian adalah berdasarkan ujian prestasi (*Performance Testing*). Jenis ujian ini adalah untuk mengenalpasti algoritma saya adalah dapat dijalankan dengan keadaan yang baik. Kaedah ini dapat menguji tentang masa responsi, kelajuan sistem, efisiensi diubah skala. Skop ujian adalah untuk mengenalpasti kerumitan algoritma saya demi komponen yang sedia ada. Ujian ini mengandungi ujian beban yang uji pada had atas dengan menambahkan dataset dan ujian stres yang uji algoritma dengan menjalankan algoritma tersebut dengan keadaan ekstreme.

5.3.1 PENGUJIAN BUKAN KEFUNGSIAN

Biasanya suatu algoritma akan mengalami masalah masa reponsi yang lama. Dalam ujian pretasi saya, saya akan cuba menguji algoritma saya dengan beberapa bahagian oleh ShanghaiTech Dataset. Bahagian A daripada dataset mengandungi kumpulan orang ramai ramai yang lebih tumpat. Manakala dalam dataset B, imej yang di dalam mengandungi kumpulan orang ramai yang kurang tumpat. Sebagai ujian stres, saya akan berbanding dengan masa yang perlu digunakan untuk kedua-dua bahagian dataset.

Selain itu, ujian masa kekuatan adalah satu bahagian yang penting untuk setiap perisian. Masa kekuatan adalah masa yang digunakan untuk perisian bermula. Masa yang impian adalah beberapa saat sahaja.

Seterusnya, ujian kejejalan (*bottleneck*) akan dijalankan supaya menguji kesilapan kod dalam algoritma saya. Kejejalan biasanya dikesankan oleh kesalahan kod yang lebih serius. Tujuan ujian adalah mencari kesalahan kod dan memperbaiki kesalahan tersebut.

5.3.2 SPESIFIKASI KES PENGUJIAN

ID	Keperluan	Penerangan
Kes 1	Part_A(Training)	Bahagian ini mengandungi 300 imej yang mempunyai kumpulan orang ramai yang lebih tumpat.
Kes 2	Part_A(Testing)	Bahagian ini mengandungi 182 imej yang mempunyai kumpulan orang ramai yang lebih tumpat.
Kes 3	Part_B(Training)	Bahagian ini mengandungi 400 imej yang mempunyai kumpulan orang ramai yang kurang tumpat.
Kes 4	Part_B(Testing)	Bahagian ini mengandungi 316 imej yang mempunyai kumpulan orang ramai yang kurang tumpat.

Jadual 5.3.2: Kes – kes pengujian

5.3.3 SPESIFIKASI PROSEDUR PENGUJIAN

1. Ujian kejejalan (*bottleneck*)

Ujian ini akan dijalankan dengan menggunakan *Static Code Anaylsis* yang sedia ada dalam Spyder 3 IDE. Selain daripada itu, algoritma akan diuji dengan menjalankan kod ini dengan beberapa kali.

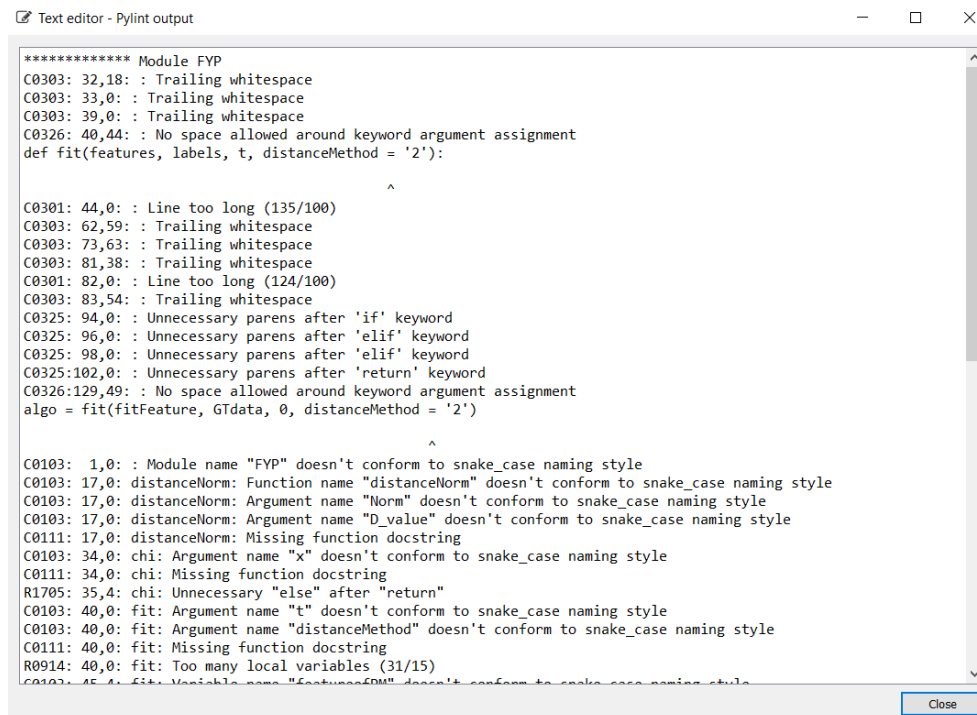
2. Ujian masa kekuatan (*Load time Test*)

Kes	Part_A (Training)	Part_A (Testing)	Part_B (Training)	Part_B (Testing)
Objektif - Ujian masa kekuatan	Untuk menguji masa kekuatan algoritma saya dari permulaan. Masa yang impian ialah beberapa saat sahaja. Semakin pendek masa kekuatan, semakin baik algoritma ini.			

Jadual 5.3.3: Ujian masa kekuatan

5.3.4 LOG PENGUJIAN

1. Ujian kejejalan (bottleneck)

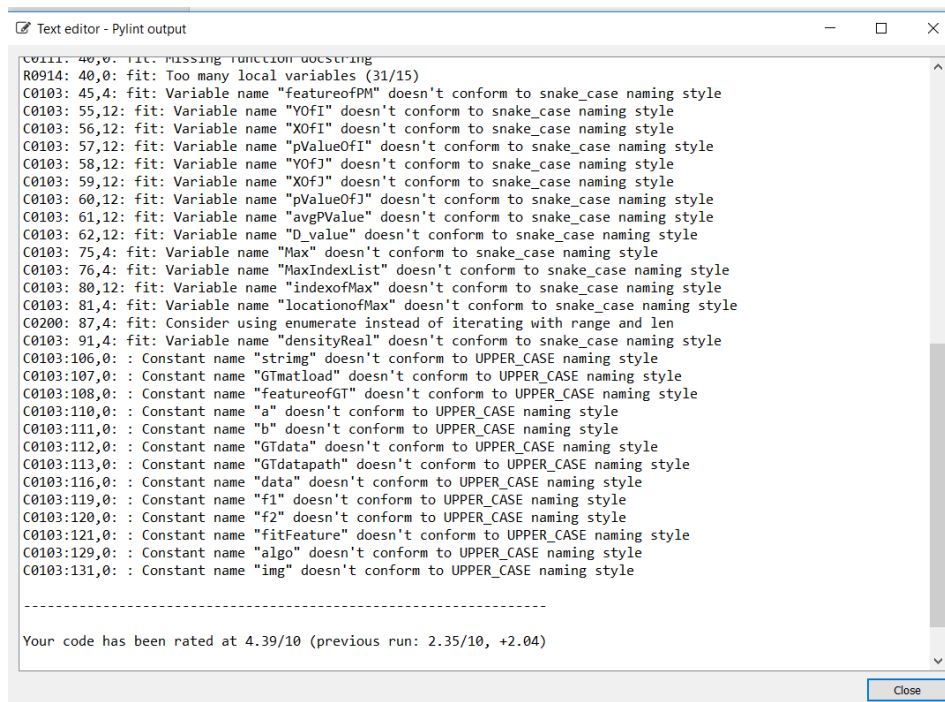


```

***** Module FYP
C0303: 32,18: : Trailing whitespace
C0303: 33,0: : Trailing whitespace
C0303: 39,0: : Trailing whitespace
C0326: 40,44: : No space allowed around keyword argument assignment
def fit(features, labels, t, distanceMethod = '2'):
    ^
C0301: 44,0: : Line too long (135/100)
C0303: 62,59: : Trailing whitespace
C0303: 73,63: : Trailing whitespace
C0303: 81,38: : Trailing whitespace
C0301: 82,0: : Line too long (124/100)
C0303: 83,54: : Trailing whitespace
C0325: 94,0: : Unnecessary parens after 'if' keyword
C0325: 96,0: : Unnecessary parens after 'elif' keyword
C0325: 98,0: : Unnecessary parens after 'elif' keyword
C0325:102,0: : Unnecessary parens after 'return' keyword
C0326:129,49: : No space allowed around keyword argument assignment
algo = fit(fitFeature, GTdata, 0, distanceMethod = '2')
    ^
C0103: 1,0: : Module name "FYP" doesn't conform to snake_case naming style
C0103: 17,0: distanceNorm: Function name "distanceNorm" doesn't conform to snake_case naming style
C0103: 17,0: distanceNorm: Argument name "Norm" doesn't conform to snake_case naming style
C0103: 17,0: distanceNorm: Argument name "D_value" doesn't conform to snake_case naming style
C0111: 17,0: distanceNorm: Missing function docstring
C0103: 34,0: chi: Argument name "x" doesn't conform to snake_case naming style
C0111: 34,0: chi: Missing function docstring
R1705: 35,4: chi: Unnecessary "else" after "return"
C0103: 40,0: fit: Argument name "t" doesn't conform to snake_case naming style
C0103: 40,0: fit: Argument name "distanceMethod" doesn't conform to snake_case naming style
C0111: 40,0: fit: Missing function docstring
R0914: 40,0: fit: Too many local variables (31/15)
C0103: 45,4: fit: Variable name "featureofPM" doesn't conform to snake_case naming style

```

Rajah 5.3.4.1: Keputusan *Static Code Analysis* – Bahagian 1



```

C0111: 40,0: fit: Missing function docstring
R0914: 40,0: fit: Too many local variables (31/15)
C0103: 45,4: fit: Variable name "featureofPM" doesn't conform to snake_case naming style
C0103: 55,12: fit: Variable name "YOOfI" doesn't conform to snake_case naming style
C0103: 56,12: fit: Variable name "XOfI" doesn't conform to snake_case naming style
C0103: 57,12: fit: Variable name "pValueOfI" doesn't conform to snake_case naming style
C0103: 58,12: fit: Variable name "YOOfJ" doesn't conform to snake_case naming style
C0103: 59,12: fit: Variable name "XOfJ" doesn't conform to snake_case naming style
C0103: 60,12: fit: Variable name "pValueOfJ" doesn't conform to snake_case naming style
C0103: 61,12: fit: Variable name "avgPValue" doesn't conform to snake_case naming style
C0103: 62,12: fit: Variable name "D_value" doesn't conform to snake_case naming style
C0103: 75,4: fit: Variable name "Max" doesn't conform to snake_case naming style
C0103: 76,4: fit: Variable name "MaxIndexList" doesn't conform to snake_case naming style
C0103: 80,12: fit: Variable name "indexofMax" doesn't conform to snake_case naming style
C0103: 81,4: fit: Variable name "locationofMax" doesn't conform to snake_case naming style
C0200: 87,4: fit: Consider using enumerate instead of iterating with range and len
C0103: 91,4: fit: Variable name "densityReal" doesn't conform to snake_case naming style
C0103:106,0: : Constant name "string" doesn't conform to UPPER_CASE naming style
C0103:107,0: : Constant name "GTmatload" doesn't conform to UPPER_CASE naming style
C0103:108,0: : Constant name "featureofGT" doesn't conform to UPPER_CASE naming style
C0103:110,0: : Constant name "a" doesn't conform to UPPER_CASE naming style
C0103:111,0: : Constant name "b" doesn't conform to UPPER_CASE naming style
C0103:112,0: : Constant name "GTdata" doesn't conform to UPPER_CASE naming style
C0103:113,0: : Constant name "GTdatapath" doesn't conform to UPPER_CASE naming style
C0103:116,0: : Constant name "data" doesn't conform to UPPER_CASE naming style
C0103:119,0: : Constant name "f1" doesn't conform to UPPER_CASE naming style
C0103:120,0: : Constant name "f2" doesn't conform to UPPER_CASE naming style
C0103:121,0: : Constant name "fitFeature" doesn't conform to UPPER_CASE naming style
C0103:129,0: : Constant name "algo" doesn't conform to UPPER_CASE naming style
C0103:131,0: : Constant name "img" doesn't conform to UPPER_CASE naming style

-----
Your code has been rated at 4.39/10 (previous run: 2.35/10, +2.04)

```

Rajah 5.3.4.2: Keputusan *Static Code Analysis* – Bahagian 2

Keputusan Ujian dengan menjalankan kod:

Tiada kesilapan kod yang serius, kod boleh dilaksanakan dengan lancar.

```
Python 3.5.6 |Anaconda, Inc.| (default, Aug 26 2018, 16:05:27) [MSC  
v.1900 64 bit (AMD64)]
```

```
Type "copyright", "credits" or "license" for more information.
```

```
IPython 6.5.0 -- An enhanced Interactive Python.
```

```
In [1]: runfile('C:/Users/Asus/Desktop/FYP/COMBINE.py',  
wdir='C:/Users/Asus/Desktop/FYP')
```

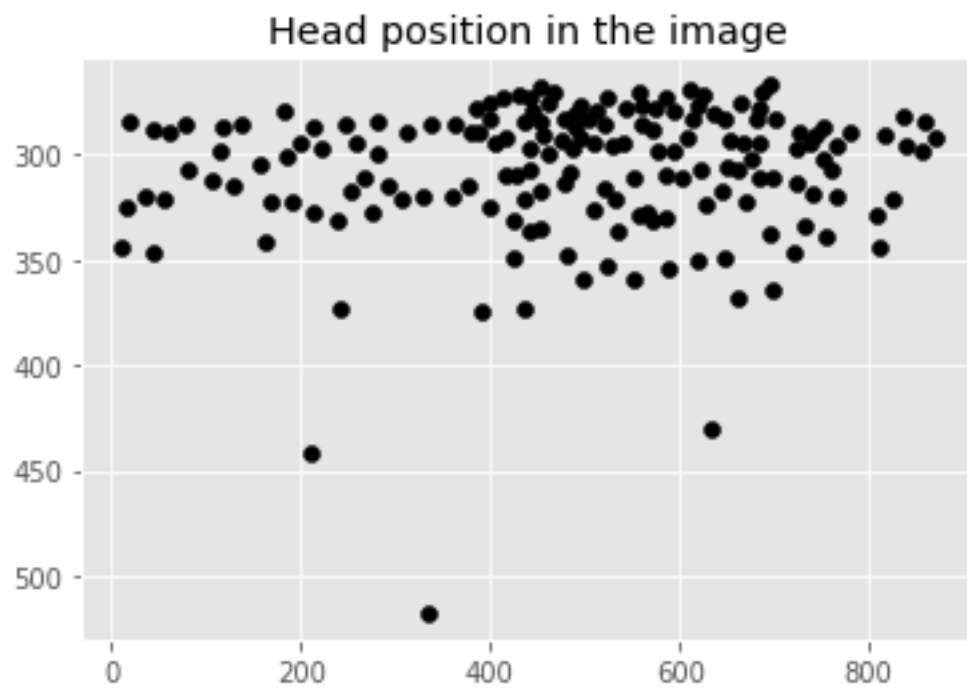
```
Please select which set of images is going to test:
```

- A. Test A
- B. Train A
- C. Test B
- D. Train B

```
A
```

```
Please insert number 1 to 182 to select image: 1
```

```
Please wait a while...
```



Please wait a while for calculation

Max Density = 17.0

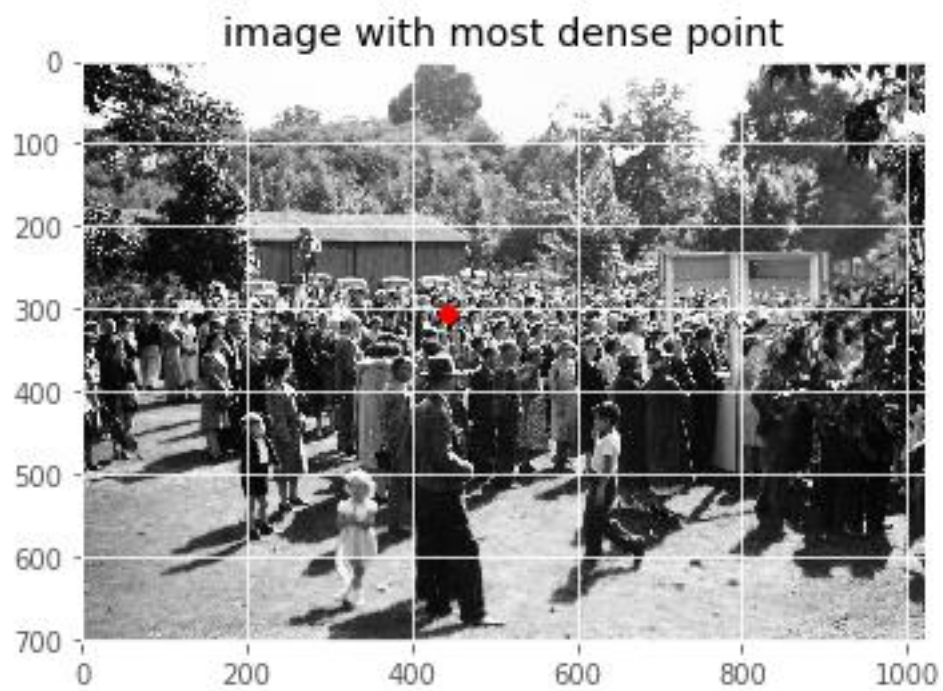
Index of Max Density Point = 56

Location for Max Density = [442.50181818181824, 306.47752247752254]

min distance for max density point = 0.6151329164489956

Density: 1.1887425471936715

Medium Dense Crowd

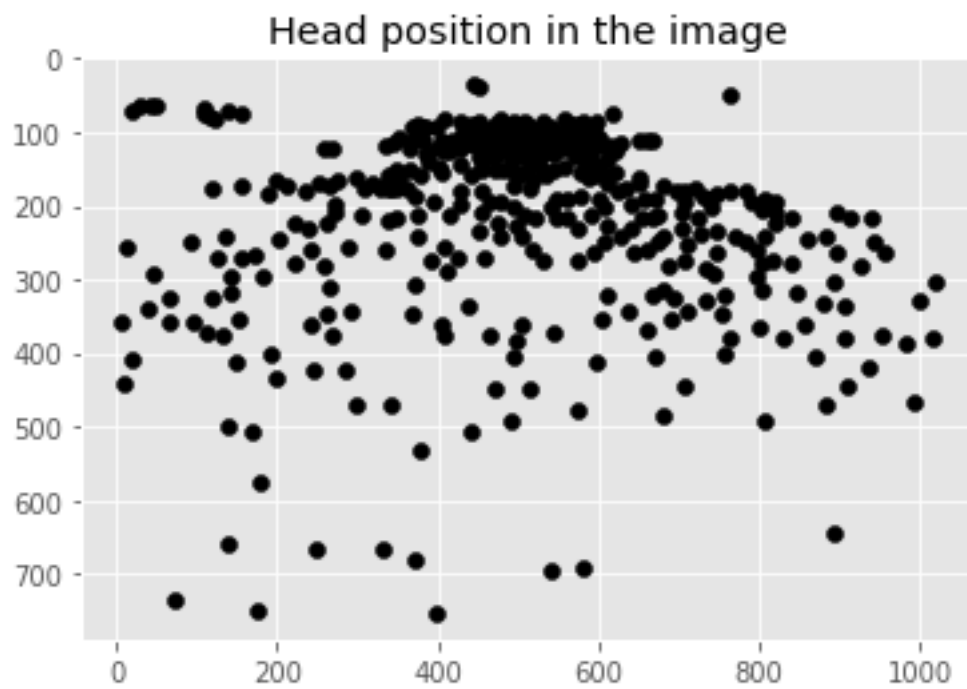


Please select which set of images is going to test:

- A. Test A
 - B. Train A
 - C. Test B
 - D. Train B
- B

Please insert number 1 to 300 to select image: 300

Please wait a while...



Please wait a while for calculation

Max Density = 39.0

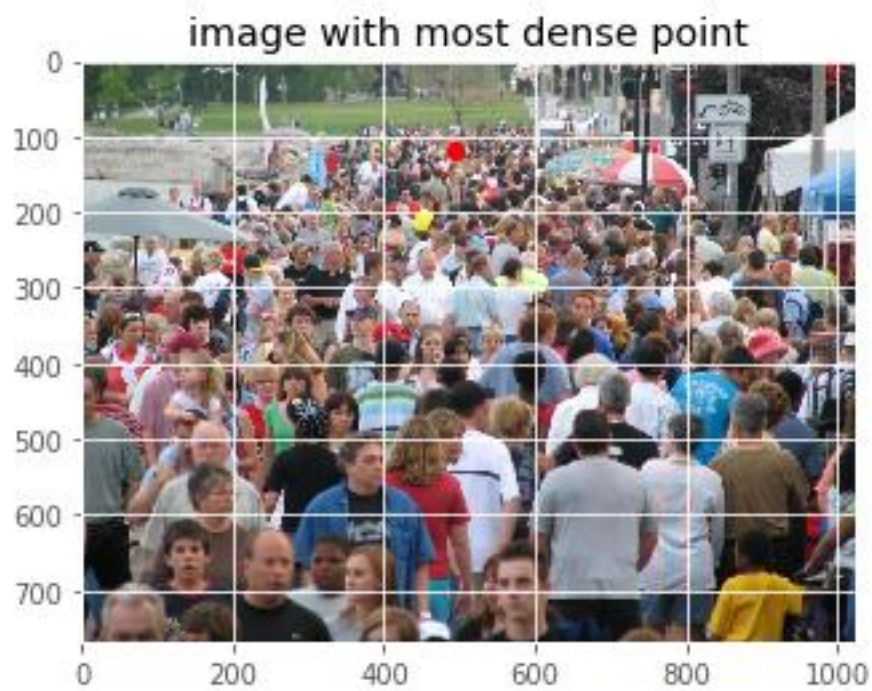
Index of Max Density Point = 168

Location for Max Density = [494.1825954198473, 119.58717557251912]

min distance for max density point = 0.4623338493859188

Density: 0.6715235610513891

Heavy Dense Crowd

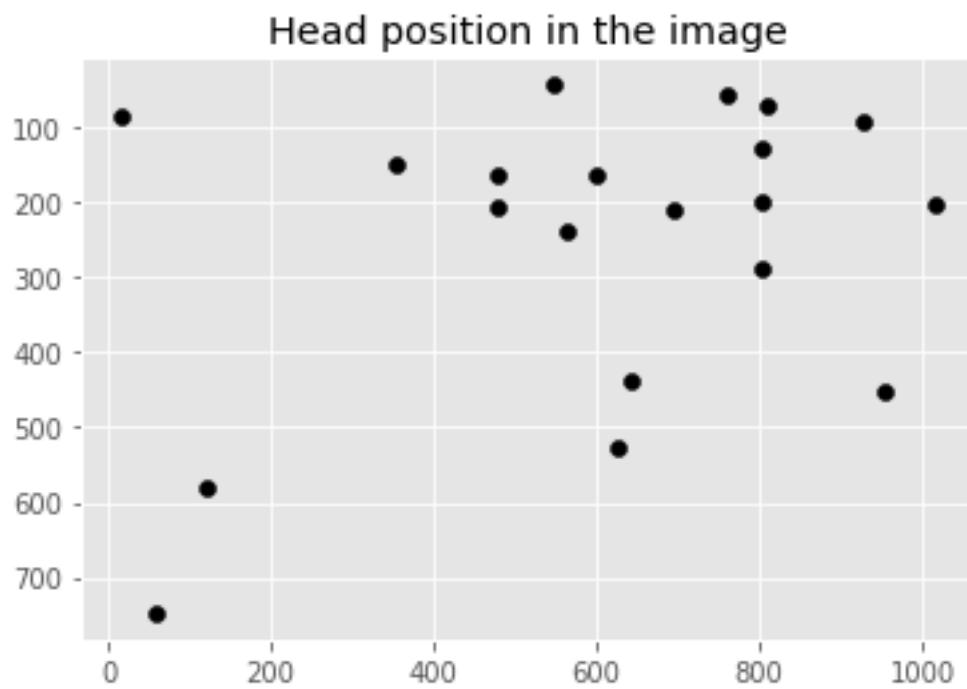


Please select which set of images is going to test:

- A. Test A
 - B. Train A
 - C. Test B
 - D. Train B
- C

Please insert number 1 to 316 to select image: 111

Please wait a while...



Please wait a while for calculation

Max Density = 4.0

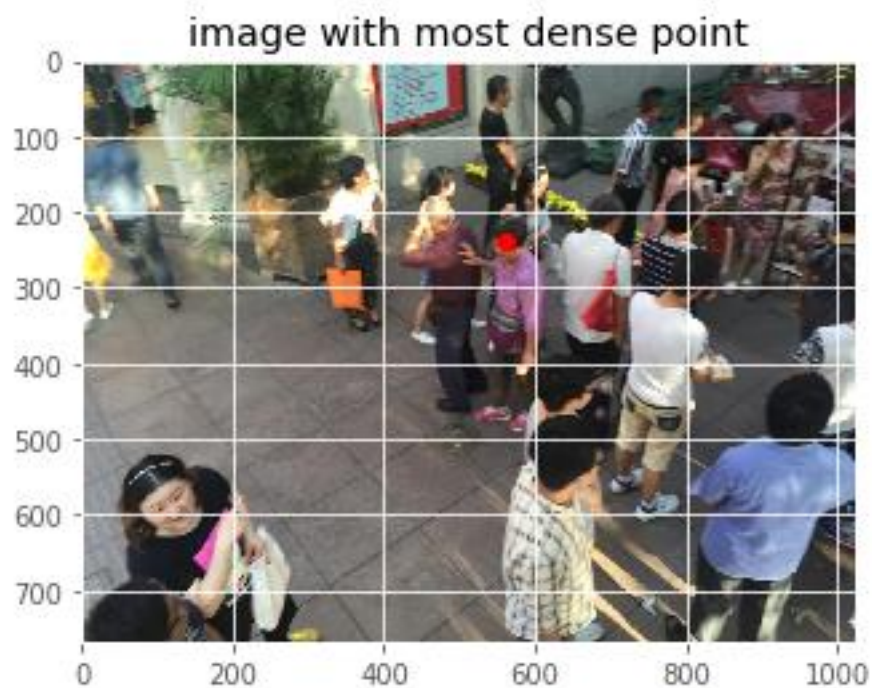
Index of Max Density Point = 10

Location for Max Density = [562.3586976074832, 237.51466091023553]

min distance for max density point = 0.7256459142127204

Density: 1.6542432882828537

Light Dense Crowd

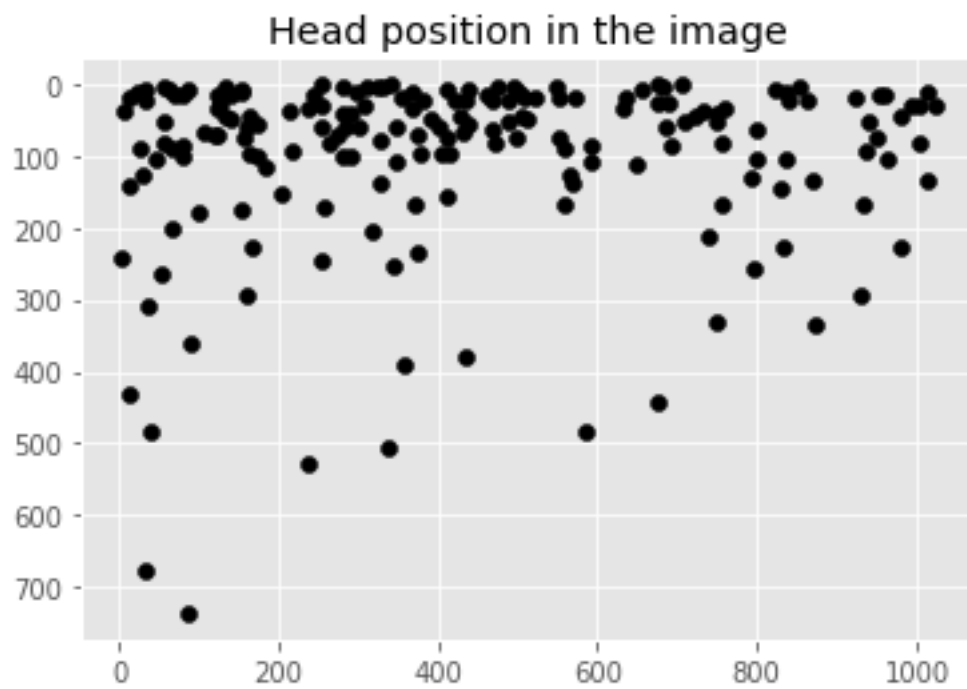


Please select which set of images is going to test:

- A. Test A
 - B. Train A
 - C. Test B
 - D. Train B
- D

Please insert number 1 to 400 to select image: 50

Please wait a while...



Please wait a while for calculation

Max Density = 0.0

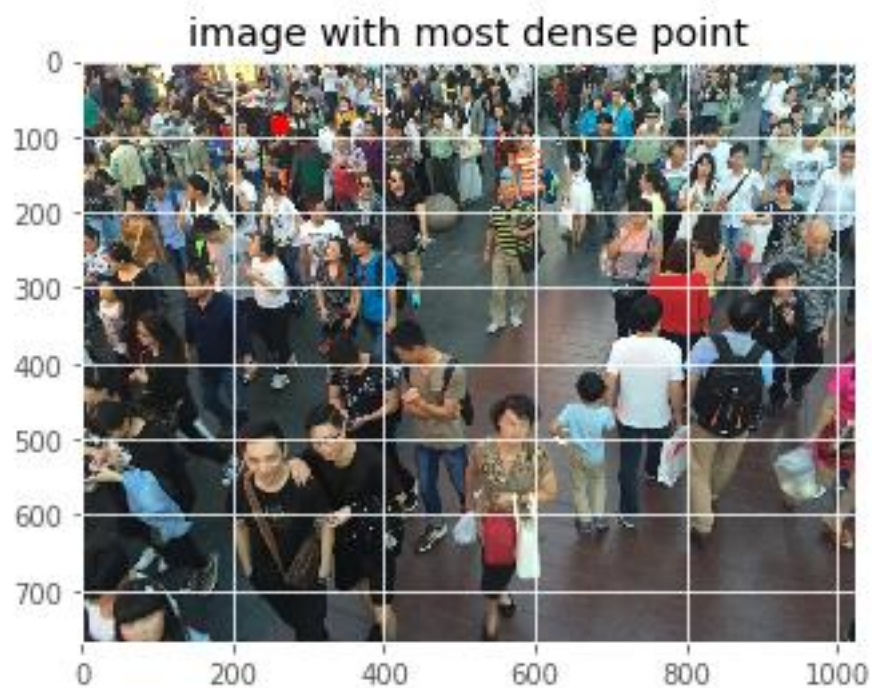
Index of Max Density Point = 184

Location for Max Density = [261.93612695914493, 82.99298314643572]

min distance for max density point = 0.35776716724590923

Density: 0.40211552174429577

Mosh-pit Crowd



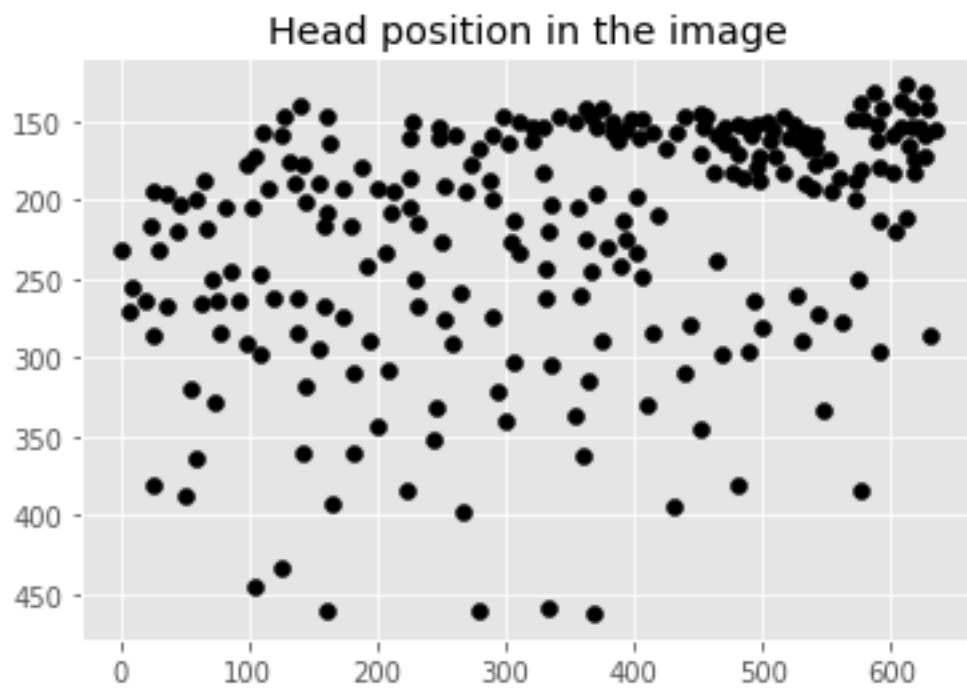
Please select which set of images is going to test:

- A. Test A
- B. Train A
- C. Test B
- D. Train B

A

Please insert number 1 to 182 to select image: 99

Please wait a while...



Please wait a while for calculation

Max Density = 49.0

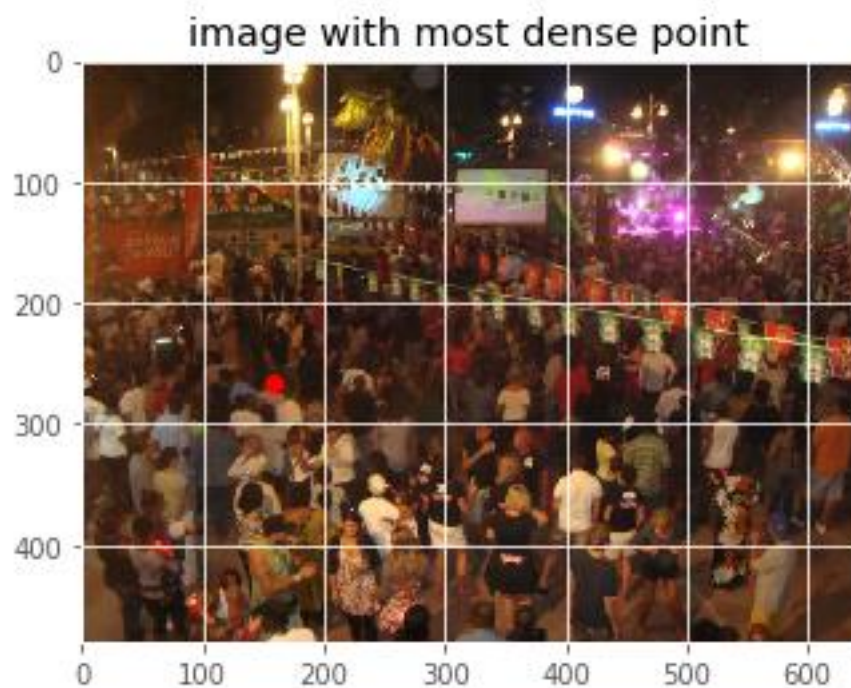
Index of Max Density Point = 52

Location for Max Density = [158.12962962962965, 266.87037037037044]

min distance for max density point = 0.6368794597980431

Density: 1.2742785063183573

Medium Dense Crowd

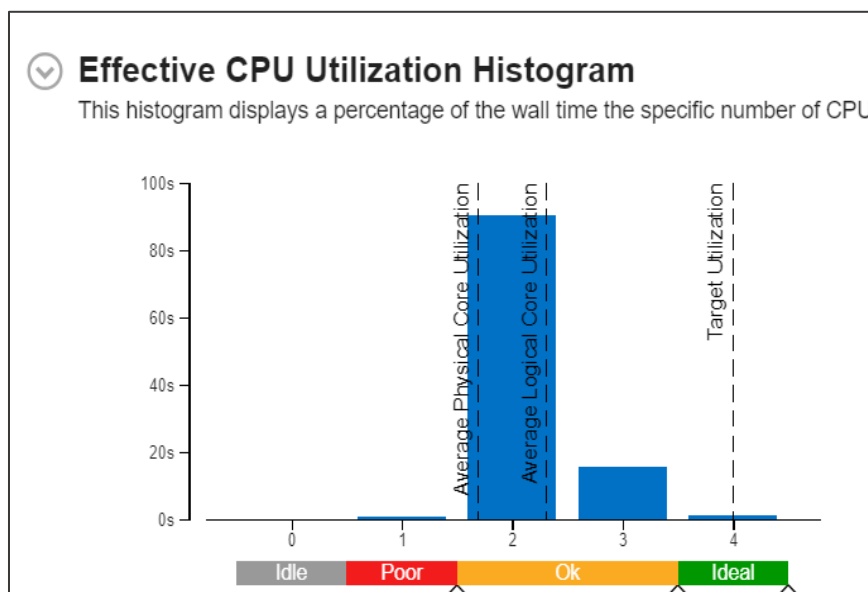


Please select which set of images is going to test:

- A. Test A
 - B. Train A
 - C. Test B
 - D. Train B
- 0

In [2]:

2. Ujian prestasi (memori, unit pemprosesan pusat)

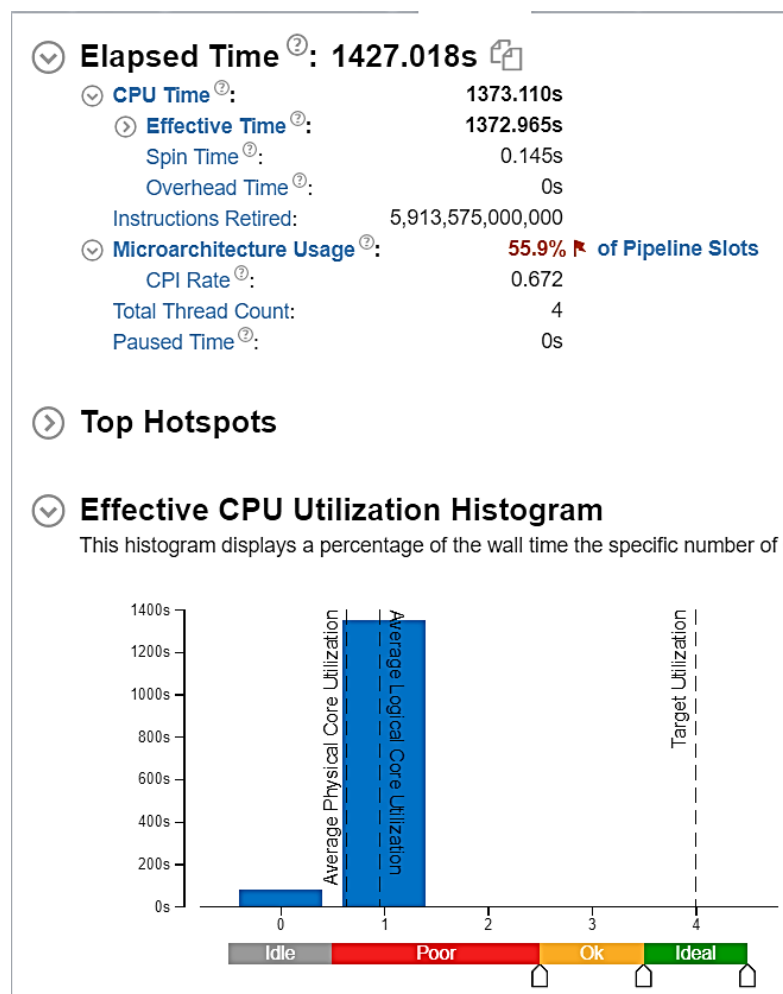


Rajah 5.3.4.3: Ujian prestasi bagi kod



Rajah 5.3.4.4: Keterangan dengan ujian prestasi

Ujian dijalankan dengan program VTune Amplifier. Ujian ini dijalankan dengan input lima imej yang berasal daripada folder yang berbeza. Dalam keputusan ujian, ia menunjuk proses pengiraan ketumpatan adalah agak baik kerana menggunakan masa yang sedikit.



Rajah 5.3.4.5: Ujian prestasi bagi kod untuk folder Test A

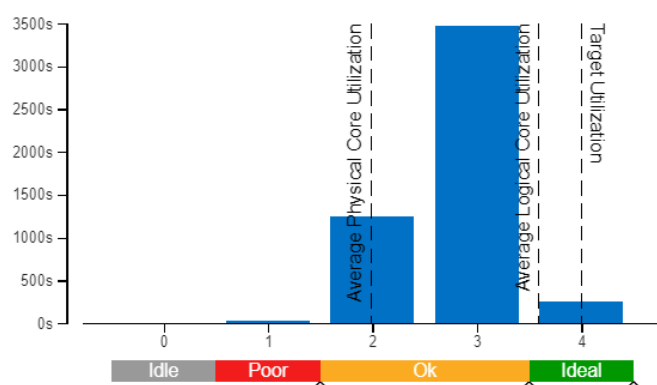
Ujian ini menjalankan kod berulang-ulangan sehingga semua imej dalam folder Test A diproseskan. Keputusan menunjukkan prosesnya agak mengambil masa yang lama kerana folder ini mengandungi 182 imej yang mempunyai kawasan yang sangat tumpat.

⌵	Elapsed Time [?]	4987.689s
⌵	CPU Time [?]	17977.739s
⌵	Effective Time [?]	17913.407s
	Idle:	0s
	Poor:	47.512s
	Ok:	16897.126s
	Ideal:	968.769s
	Over:	0s
	Spin Time [?] :	64.333s
	Overhead Time [?] :	0s
	Instructions Retired:	47,812,250,000,000
⌵	Microarchitecture Usage [?]	46.6% of Pipeline Slots
	Total Thread Count:	3,921
	Paused Time [?] :	0s

➤ Top Hotspots

⌵ Effective CPU Utilization Histogram

This histogram displays a percentage of the wall time the specific number of CPU:



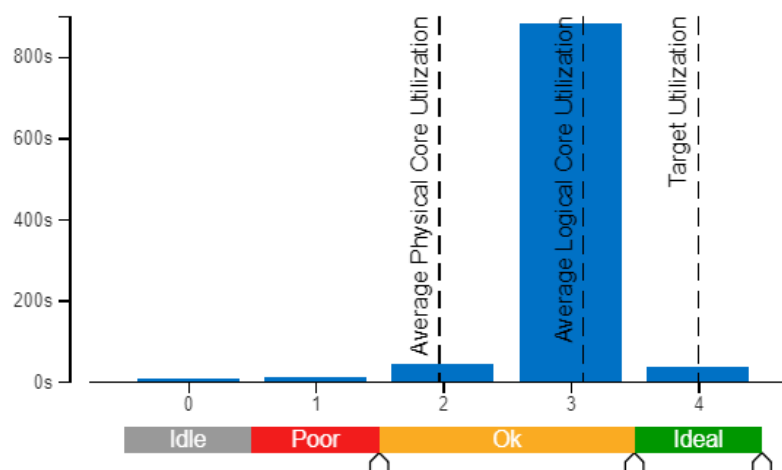
Rajah 5.3.4.6: Ujian prestasi bagi kod untuk folder Train A

Ujian ini menjalankan kod berulang-ulangan sehingga semua imej dalam folder Train A diproseskan. Keputusan menunjukkan prosesnya agak mengambil masa yang lama kerana folder ini mengandungi 300 imej.

⌵ Elapsed Time [?] :	972.222s
⌵ CPU Time [?] :	3019.039s
⌵ Effective Time [?] :	3013.561s
Idle:	0.106s
Poor:	10.176s
Ok:	2861.748s
Ideal:	141.532s
Over:	0s
Spin Time [?] :	5.478s
Overhead Time [?] :	0s
Instructions Retired:	10,692,700,000,000
⌵ Microarchitecture Usage [?] :	51.0% of Pipeline Slots
Total Thread Count:	755
Paused Time [?] :	0s

⌵ **Effective CPU Utilization Histogram**

This histogram displays a percentage of the wall time the specific number of



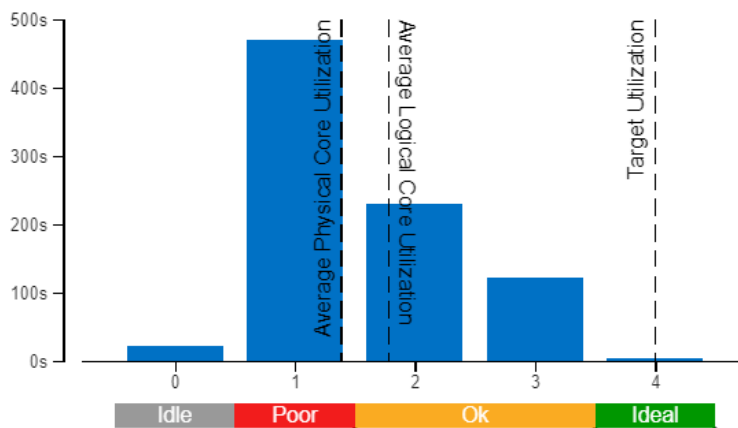
Rajah 5.3.4.7: Ujian prestasi bagi kod untuk folder Test B

Ujian ini menjalankan kod berulang-ulangan sehingga semua imej dalam folder Test B diproseskan. Keputusan menunjukkan prosesnya agak mengambil masa yang lama sederhana walaupun folder ini mengandungi 316 imej. Imej yang di dalam kurang mempunyai kawasan yang sangat tumpat.

Elapsed Time [?]	849.931s
CPU Time [?]	1523.090s
Effective Time [?]	1514.564s
Idle:	5.768s
Poor:	537.076s
Ok:	959.095s
Ideal:	12.625s
Over:	0s
Spin Time [?]	8.526s
Overhead Time [?]	0s
Instructions Retired:	6,090,575,000,000
Microarchitecture Usage [?]	44.4% ▴ of Pipeline Slots
Total Thread Count:	652
Paused Time [?]	0s

Effective CPU Utilization Histogram

This histogram displays a percentage of the wall time the specific number of CF



Rajah 5.3.4.8: Ujian prestasi bagi kod untuk folder Train B

Ujian ini menjalankan kod berulang-ulangan sehingga semua imej dalam folder Test B diproseskan. Keputusan menunjukkan prosesnya agak mengambil masa yang lama sederhana walaupun folder ini mengandungi 400 imej. Imej yang di dalam kurang mempunyai kawasan yang sangat tumpat seperti Test B.

5.4 KESIMPULAN

Bab ini telah menjelaskan pembangunan dan pengujian bagi algoritma saya. Proses pembangunan telah diuraikan. Pengujian algoritma telah dijalankan supaya algoritma ini dapat dijalankan dengan lancar dan tidak menambah beban pengkomputeran.

BAB VI

KESIMPULAN

Anggaran dan kefahaman orang ramai adalah teknologi yang penting untuk mengekalkan keselamatan awam dan melaksanakan kawalan orang ramai. Terlalu tumpat di suatu kawasan akan menyebabkan banyak kematian dan kecederaan apabila berlaku rempuhan manusia. Semua kemalangan ini boleh dielakkan apabila kumpulan orang ramai dikesan terlalu ramai di suatu tempat. Dalam kajian ini, saya telah menggunakan perkelompokan puncak ketumpatan (*Density Peak Clustering*) untuk membantu saya dalam kefahaman kumpulan orang ramai. Nilai perspektif dari output PACNN dan nilai kebenaran tanah dari ShanghaiTech dataset akan digunakan sebagai dataset saya. Kawasan kumpulan orang ramai yang “mosh-pit” tumpat akan dikesan melalui algoritma perkelompokan puncak ketumpatan dengan mengira ruang antara orang ramai. Output projek ini ialah anggaran orang ramai dan ketumpatan maxima orang ramai tersebut. Senarai koordinat dan kawasan orang ramai yang “mosh-pit” tumpat akan dijana dan imej orang ramai ditandakan dengan kawasan yang “mosh-pit” tumpat akan dihasilkan.

Had bagi projek ini adalah ia tidak dapat mengesan keadaan masa nyata kerana ia menganalisis imej, bukan video. Masalahnya bagi projek ini ialah keperluan untuk menyediakan model ini adalah agak tinggi.

Selepas projek ini, model boleh digunakan untuk melaksanakan sistem sebenar. Semasa suatu acara, bilik kawalan boleh menggunakan model ini untuk mengawasi jika kumpulan orang ramai terlalu tumpat bagi mengelakkan rempuhan manusia. Cadangan untuk menyempurnakan projek ini adalah menggunakan PACNN dalam sistem supaya dapat menganalisis imej terus, bukan dapatkan data daripada kebenaran tanah. Penyihiran pada masa hadapan bagi projek ini adalah untuk mengesan jenis kumpulan orang ramai kerana jenis orang ramai juga memainkan peranan yang penting untuk pengawalan kumpulan orang ramai.

BAB VII

RUJUKAN

- A. B. Chan and N. Vasconcelos. 2009. Bayesian Poisson regression for crowd counting. *In ICCV, pages 545–551. IEEE.*
- A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos. 2008. Privacy preserving crowd monitoring: counting people without people models or tracking. *In CVPR, pages 1–7. IEEE.*
- Alex Rodriguez, Alessandro Laio. 2014. Clustering by fast search and find of density peaks. <https://science.sciencemag.org/content/344/6191/1492.full>, 27 Jun 2014.
- Andrea Trevino. 2016. LEARN DATA SCIENCE, MACHINE LEARNING, Introduction to K-means Clustering. Oracle + DataScience.com. <https://www.datascience.com/blog/k-means-clustering>. December 6, 2016
- Anthony C. Davies, Jia Hong Yin, Sergio A. Velastin. 1995. Crowd monitoring using image processing. *Electronics & Communication Engineering Journal*, 1995.
- D. Helbing and P. Mukerji. 2012. Crowd disasters as systemic failures: Analysis of the Love Parade Disaster. <http://www.epjdatascience.com/content/1/1/7/abstract>. *EPJ Data Science* 1:7 (2012), {{Doi:10.1140/epjds7}}
- Deeksha S. 2016. Crowd: Meaning, Types and Characteristics | Psychology. <http://www.psychologydiscussion.net/social-psychology-2/crowd-behaviour/crowd-meaning-types-and-characteristics-psychology/1356>.
- Dillion Cariveau. 2015. Crowd Size Estimation. <http://course1.winona.edu/cmalone/promotion/UndergraduateResearch/Cariveau%20Report.pdf>.
- Funda Durupinar, Uğur Gökükbay, Aytok Aman, Norman I. Badler. 2015. Psychological Parameters for Crowd Simulation: From Audience to Mobs.
- Helbing. 2012. Why Crowds Can Turn Deadly. <https://www.citylab.com/equity/2012/07/physics-explain-deadly-crowd-disasters/2550/>. Jul 12, 2012.

- Kelvin Salton do Prado. 2017. How DBSCAN works and why should we use it?
<https://towardsdatascience.com/how-dbscan-works-and-why-should-i-use-it-443b4a191c80>. Apr 1, 2017.
- Lingke Zeng, Xiangmin Xu, Bolun Cai, Suo Qiu, Tong Zhang. 2017. Multi-scale Concolutional Neural Networks for Crowd Counting. School of Electronic and Information Engineering South China University of Technology, Guangzhou, China. *arXiv:1702.02359v1 [cs.CV]*. 8 Feb 2017.
- MA Security Group. 2017. CROWD COUNTING – A CURIOUS SCIENCE.
<https://maservicesgroup.com.au/crowd-counting-a-curious-science/>. August 23, 2017.
- Miaojing Shi, Zhaohui Yang, Chao Xu, Qijun Chen. 2018. Perspective-Aware CNN For Crowd Counting. *Arxiv Sanity Preserver*. 5 Jul 2018.
- Mike Rothschild. 2015. The Worst Human Stampedes.
<https://www.ranker.com/list/worst-human-stampedes-092415/mike-rothschild>.
- Okoli, Al Chukwuma, and Nnorom, Kingsley C. April, 2014. Disaster Risks in Crowded Situations: Contemporary Manifestations and Implications of Human Stampede in Nigeria. *Internationalsal Journal of Liberal Arts and Social Science*, Vol 2, No. 3.
- P. Viola, M. J. Jones, and D. Snow. 2005. Detecting pedestrians using patterns of motion and appearance. *International Journal of Computer Vision*, 63(2):153–161.
- V. Rabaud and Serge Belongie. 2006. Counting crowded moving objects. *In CVPR, volume 1, pages 705–711. IEEE, 2006*.
- Waffa M.Shalash, Aliaa Al Hazimi, Basma Al Zahrani. 2017. A Mobile Based Crowd Management System. *International Journal of Advanced Research in Computer and Communication Engineering*.
- Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, Yi Ma. 2016. Single-Image Crowd Counting via Multi-Column Convolutional Neural Network. Shanghai University.