

KEFAHAMAN KUMPULAN ORANG RAMAI DENGAN ALGORITMA PERKELOMPOKAN PUNCAK KETUMPATAN

Van Tze Shan

Dr. Hadi Affendy Dahlan

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.

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. Kesusakan 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. Mukerji 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 ramai yang penggubahan 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 IIARCCE. (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)

2. PENYATAAN 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 dihimpit 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 Güdükbay, 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.

3. OBJEKTIF KAJIAN

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 mengimplikasi nilai perspektif.

4. METOD KAJIAN

4.1. Fasa Perancangan

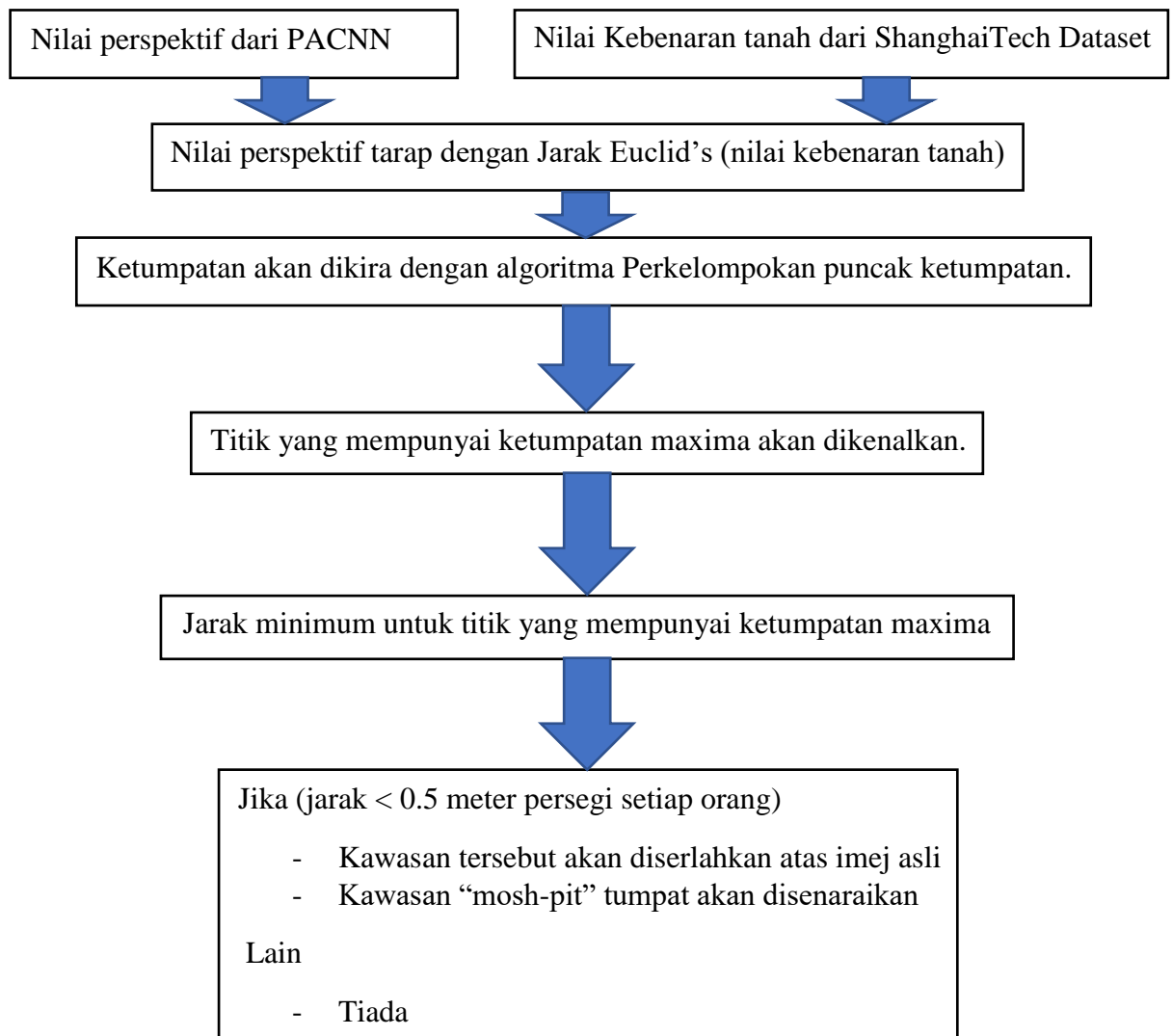
Anggaran kumpulan orang ramai dan kefahaman kumpulan orang ramai adalah teknik yang penting untuk mengekalkan keselamatan awam dan melaksanakan kawalan kumpulan. Apabila tempat terlalu sesak, rempuhan manusia akan berlaku dan menyebabkan kematian dan luka yang tidak terkira banyaknya. Kefahaman kumpulan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

4.2. Fasa Analisis

Dalam fasa ini, kajian tentang nilai perspektif daripada PACNN didarapkan dengan pengiraan jarak dijalankan. Kajian ini adalah untuk memastikan jarak yang didapati adalah jarak yang sebenar. Selain itu, untuk mendapatkan ketumpatan orang ramai, perkumpulan ketumpatan puncak (DPC) telah diimplikasikan dan diubahsuaikan. DPC ini digunakan kerana ia dapat mencari pusat titik. Teorinya ialah pusat titik biasanya merupakan titik yang paling tumpat. Oleh itu, DPC digunakan supaya dapat mengetahui pusat titik dan juga ketumpatannya.

4.3. Fasa Reka Bentuk



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.

4.4. Fasa Pengujian

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.

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.


```
Text editor - Pylint output

***** 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
C0103: 55,12: fit: Variable name "YOfI" 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 "YOfJ" 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 4.4.1: Keputusan *Static Code Analysis* – Bahagian 1

```
Text editor - Pylint output

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 "YOfI" 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 "YOfJ" 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)
```

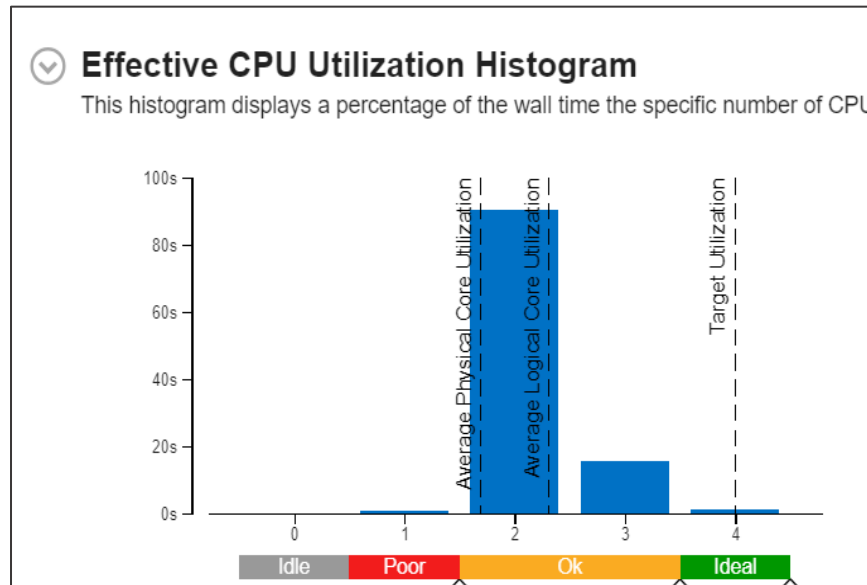
Rajah 4.4.2: Keputusan *Static Code Analysis* – Bahagian 2

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

Kes	Part_A (Training)	Part_A (Testing)	Part_B (Training)	Part_B (Testing)
-----	-------------------	------------------	-------------------	------------------

Objektif-Ujian masa kemuatan	Untuk menguji masa kemuatan algoritma saya dari permulaan. Masa yang impian ialah beberapa saat sahaja. Semakin pendek masa kemuatan, semakin baik algoritma ini.
------------------------------	---

Jadual 4.4.2: Ujian masa kemuatan



Rajah 4.4.3: Ujian prestasi bagi kod



Rajah 4.4.4: 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.

Folder	Test A	Train A	Test B	Train B
Masa Laluan (s)	1427.018	4987.689	972.222	849.931
Masa unit pemprosesan pusat (s)	1373.110	17977.739	3019.039	1523.090

Jadual 4.4.3: Perbandingan keputusan ujian prestasi mengikut folder

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.

Keputusan imej dalam folder Train A menunjukkan prosesnya agak mengambil masa yang lama kerana folder ini mengandungi 300 imej.

Bagi imej dalam folder Test B, Keputusan ujian menunjukkan prosesnya agak mengambil masa yang lama sederhana walaupun folder ini mengandungi 316 imej. Imej yang di dalam kurang mempunyai kawasan yang sangat tumpat.

Ujian ini juga menjalankan kod bagi imej dalam folder Test B. 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. HASIL KAJIAN

5.1. Hasil kod

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.

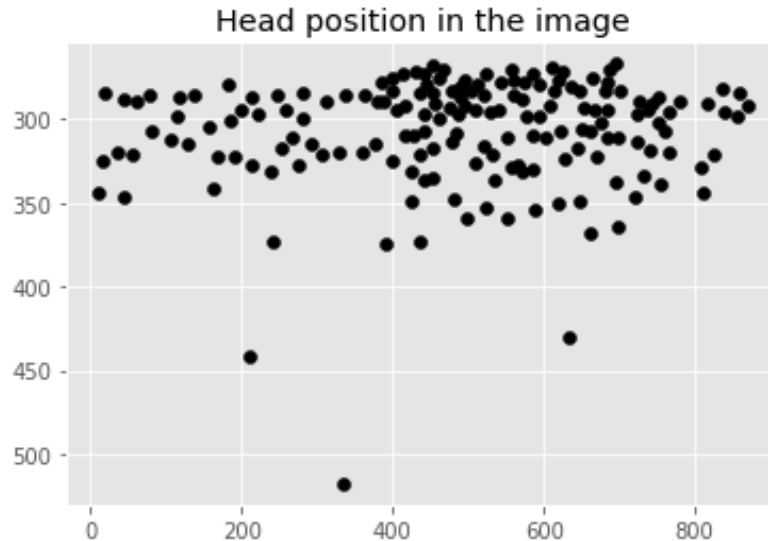
```
In      [1]:      runfile('C:/Users/Asus/Desktop/FYP/COMBINE.py',  
wdir='C:/Users/Asus/Desktop/FYP')
```

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

Rajah 5.1: Hasil kod – Bahagian 1

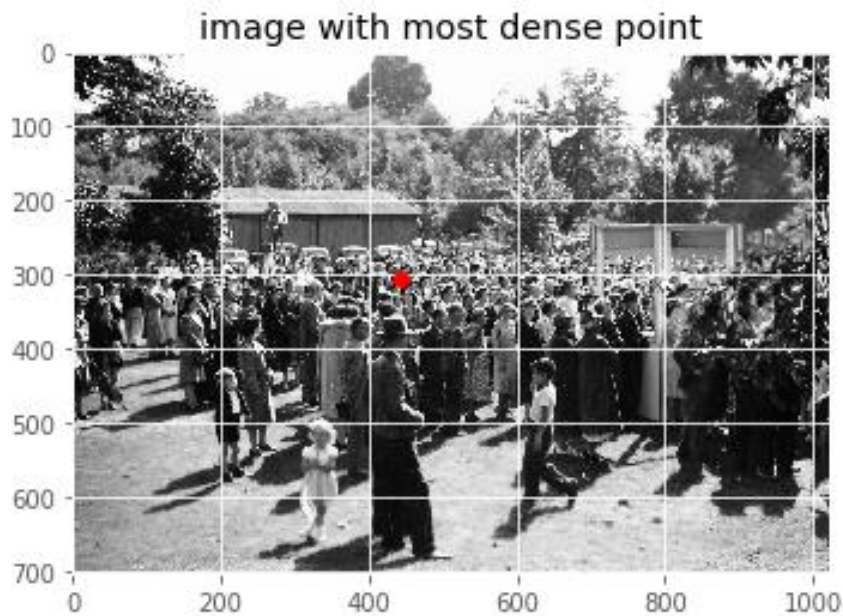
Index of Max Density Point = 56

Location for Max Density = [442.50181818181824, 306.47752247752254]

min distance for max density point = 0.6151329164489956

Density: 1.1887425471936715

Medium Dense Crowd



Rajah 5.2: Hasil kod – Bahagian 2

Hasil projek ini adalah seperti Rajah 5.1 dan Rajah 5.2. Kod ini akan dijalankan berulang – ulangan sehingga input pengguna selain daripada A, B, C atau D apabila memilih folder.

5.2. Hasil berdasarkan Dataset Shanghai

Jenis Orang Folder \ Ramai	Kurang Tumpat	Sederhana Tumpat	Sangat Tumpat	Paling Tumpat	Jumlah
Test A	41	47	62	32	182
Train A	55	74	112	59	300
Test B	36	20	78	182	316
Train B	58	35	85	222	400

Jadual 5.2.1: Keputusan kod atas imej daripada Dataset Shanghai.

6. 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.

7. 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 Durupınar, Uğur Gökükbay, Aytek 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. ShanghaiTech University.