Crowd Congestion Assessment using Multi-Resolution Clustering

Yasser Mohammad Seddiq, Ayman A. Alharbiy, Moayyad Hamza Ghunaim
King Abdulaziz City for Science and Technology

تقييم ازدحام الحشود باستخدام التكتل ذي الدقة المتغيرة

ياسر بن محمد صديق، أيمن بن أحمد الحربي، مؤيد بن حمزة غنيم
مدينة الملك عبدالعزيز للعلوم و التقنية

ملخص البحث (Abstract):

يتطرق هذا البحث لمشكلة ازدحام الحشود و التي تحدث في عدة مواقع في الحج و العمرة كمداخل المسجد الحرام و جسر الجمرات من خلال حل يعتمد على تقنية التكتل، فعند التقاط صورة علوية للحشود المراد تحليلها يبدأ الخوارزم بإيجاد الكتل ذات الكثافة العالية التي تتسم بتقارب أعضاء الكتلة الواحدة بمسافة لاتزيد عن حد معين، و بما أن الازدحام يحصل دائما في الكتل البشرية الضخمة، فإن الكتل التي يستخرجها الخوارزم تفرز حسب أحجامها حيث تستبعد الكتل الصغيرة بينما تخضع الكتل الكبيرة لمزيد تحليل و معالجة، و لأن ثمة احتمال بأن تكون إحدى تلك الكتل الكبيرة ما هي إلا تجمع لكتل أصغر تبدو كما لو كانت كتلة واحدة، فإن الكتل الكبيرة المستخرجة في المرحلة السابقة تخضع لعملية تكتل مرة أخرى و لكن باعتبار مسافة أقرب بين أعضاء الكتلة الواحدة و بهذه الطريقة فإن الكتل الصغيرة التي كانت تميز بأنها كتلة واحدة مندمجة ستنفصل عن بعضها البعض و من ثم تستبعد، بينما الكتل الكبيرة ستصمد أمام محاولة تقسيمها فيميزها الخوارزم بأنها كتل واحدة مزدحمة، ثم يتم أخيرا تقييم درجة الازدحام إلى ثلاث درجات: أشد درجة حينما توجد الكتل الكبيرة في كلتا المرحلتين، أما أوسطها فحين تظهر الكتل الكبيرة في المرحلة الأولى ثم تختفي في الثانية، و حين لا تظهر الكتل الكبيرة عند المرحلة الأولى فتلك أقل درجات الزحام.

This work addresses the problem of crowd congestion that could happen in many places in the Hajj and Umrah such as at the gates of the Haram or the Jamaraat Bridge. The proposed solution utilizes unsupervised clustering technique that is applied on a top view image of the crowd. The first stage of the algorithm determines the number of clusters whose members are within proximity threshold. Based on the fact that large clusters are likely to represent a congestion, clusters are qualified upon their sizes where small clusters are disqualified as congestion candidates. However, it is still possible that a big cluster might consist of smaller clusters that are recognized as one super-cluster due to the coarse resolution of the proximity threshold applied in the first stage. Therefore, a potential congested cluster is further analyzed using a proximity threshold of finer granularity to examine its steadfastness against sub-clustering. Finally, based on the outcomes of the two stages, the degree of congestion is assessed into three levels: the highest degree is when big clusters appear in the first stage and persist in the second, the moderate degree is when big clusters appear in the first stage, but disappear in the second, whereas the lowest degree is when no big cluster appear at all. The algorithm is tested under different scenarios of various degrees of congestion.

Introduction:

Crowd congestion is a critical challenge during Hajj and Umrah. Authorities practice extreme caution observing the potential places of congestion such as gates of The Grand Mosque, Jamaraat Bridge and walking paths. Well-trained officials monitoring crowds are capable to spotting potential congestions early enough to avoid any undesired consequences. Providing those officials with smart assistive technologies would significantly boost their efficiency and enable them to solve more complicated problems in the field.

Clustering algorithms are efficient data mining tools that can be used in crowd management applications. Clustering algorithms differ on their dependence on crowd shape and centroid [1]. Thus, algorithms that are shape- and centroid-independent are the most suitable for crowd management applications due to the high agility of the shape of a crowd mass and its infinite shape possibilities.

Clustering algorithms do the job of grouping nodes in that are logically related to each other. In a crowd management problem, nodes relationships are checked based on distance such that a nodes that are within proximity of each other are deemed one cluster. Clustering algorithms are considered of NP-hardness [2]. They can be classified into two categories: (i) partitional clustering where nodes are divided into non-overlapping clusters, and (ii) hierarchical clustering, where the algorithm divides big clusters into smaller ones (top-down) or merges small clusters into bigger ones (bottom-up) [3—5]. Both types can serve in solving crowd management problems.

Research aims:

This paper presents reports the work of devising an algorithm capable of detecting a congestion crowd and evaluating its severity level. The algorithm involves multiresolution clustering that enables the algorithm to distinguish the density of a crowd mass. Congestion is measured based on three-level evaluation scale. The paper presents detailed description of the algorithm and its test results.

Research methodology:

The proposed algorithm measures the severity of a crowd congestion by applying two stages of clustering on the crowd scene. The algorithm involves two clustering stages: a coarse-resolution clustering roughly spotting the zones of potential congestion followed by a fine-resolution re-clustering stage focusing on those hot zones to check their persistence. By the second clustering stage, if all nodes within the hot zone clusters are busted into individual nodes that are too sparse or too few to form one cluster, then, this crowd is deemed moderately congested. The more persistent the nodes to stay in one cluster after the double clustering, the higher the congestion severity is estimated. A safe situation is declared either when no hot zones are detected by the coarse clustering stage (hence no need to start the fine clustering), or when there is initially no enough nodes to process by the coarse clustering stage in the first place. The detailed steps of the multi-resolution algorithm is described in the following.

The flow chart of Figure 1 illustrates the algorithm. A preprocessing stage is assumed that involves acquiring a top-view image of the crowd and extracting the (x,y)-coordinates of people in the scene. The algorithm starts by counting the people making the crowd. Unless the number of people exceeds threshold N, the algorithm does not proceed and a "no

congestion" decision is made terminating the process. Otherwise, the second step of clustering is applied on the data considering coarse distance resolution. At this resolution level, any two persons separated by distance less then r are considered members of the one cluster. The outcome of the coarse-resolution clustering is a set of clusters formed by people who are in close proximity of each other. Among those clusters, there could be small clusters containing too few people that do not give signs of potential congestion. Such clusters are eliminated and discarded in the subsequent steps. The remaining big clusters, which are called the hot zones, are worth worrying about and, hence, are qualified for further re-clustering. In case when no hot zones are identified by the coarse-resolution clustering, the algorithm terminates by deciding "no congestion". Otherwise, the next step of re-clustering the hot zones starts.

The second step involves fine-resolution clustering, where people falling into the hot zone are grouped together based on distance threshold d such that d < r. The value gained by using finer distance resolution is to discover whether some hot zone is really formed by congested peopled or it is just a super-cluster made of smaller clusters. In this stage, the hot zones are processed one by one. The outcome of each iteration is also a set of clusters forming the hot zone under test. Similar to the previous step, the small clusters are disqualified. If all clusters within the current hot zone are disqualified, then, the current iteration ends and the next iteration starts by re-clustering the next hot zone. The algorithm proceeds tackling all hot zones, and if all busted, the algorithm declares "moderate congestion" and terminates.

If it happens that in one of the iteration described above, the re-clustering process returns at least one yet qualified cluster, then, the hot zone persists and the algorithm fails to bust that hot zone into negligible sub-clusters. In this case, the algorithm dose not iterate any further, but terminates by deciding "severe congestion".

While the fine-resolution clustering alone is adequate to detect congestion or no-congestion states, it fails to evaluate the level of severity whether it is severe or moderate as the multi-resolution approach does.

Results and discussion:

The algorithm is applied on synthetic data that is generated to mimic different crowd scenarios as depicted in Figure 2. Result plots are presented in the figure as a 2D matrix where the left-most column shows the original data before being input to the algorithm, the middle column shows the output of the coarse-resolution clustering stage, and, finally, the right-most column shows the output of the fine-resolution clustering stage. Each row of the plot is a separate case being processed by the algorithm.

The first crowd case is shown in the first row of the figure (Figure 2(a)–(c)). This is a top-view of a moderate-congestion randomly scattered crowd. The coarse clustering stage manages to bust some nodes that are missing in Figure2(b), but it still can recognize data as five big clusters (hot zones). When the algorithm tests the steadfastness of the hot zones against sub-clustering, they all disappeared. Hence, this is evaluated as a moderate congestion.

Two different no-congestion crowd cases are presented in Figure 2(d)–(f) and Figure 2(g)–(i). The nodes, in both cases, cannot stand the coarse-resolution clustering as can be observed in the empty plots in the middle. Thus, those are declared as "no congestion".

Figure 2(j)–(l) illustrates the results of a crowd scene of people organized in groups and queues of several sizes and shapes. The two tiny groups on the top of the scene are filtered out by the coarse clustering stage. Later in the fine-resolution clustering, the rest of the groups disappear, which demonstrates a moderately congested crowd. Likewise, Figure 2(m)–(o) illustrates the results of a crowd scene of people in a walking path. The algorithm also evaluates the case as a moderate-congestion crowd.

Finally, the crowd depicted in Figure(p)–(r) mimics a severely congested crowd of people moving throw a wide path getting narrower at the end. It can be clearly seen that the scattered sparse nodes disappear by the coarse-resolution stage. Also, some of the remaining nodes are filtered out by the second stage, while the major crowd mass is persistent till the end. Therefore, this is evaluated as a severe-congestion crowd.

The discussion above demonstrates the correctness of the algorithm and its quality to serve as computer-aided assistive tool for officials who watch crowd and helping them to make the right decision in the proper time.

Summary and conclusion:

The paper reported the work of devising an algorithm that detects crowd and evaluates level of congestion in three-level scale: no congestion, moderate congestion, and severe congestion. The algorithm is capable of evaluating congestion severity by adopting a multiresolution clustering approach. Input data are subject to clustering stages at coarse-resolution and fine-resolution, respectively. The algorithm is implemented and test using synthetic data mimicking crowd scenes of different severity levels. The Algorithm showed correctness in processing them and made evaluation decisions matching expectations.

Recommendations:

3. The research demonstrated the usefulness of the multiresolution clustering in evaluating crowd congestion severity level.
4. The algorithm has a great potential in serving in assistive technologies and computer-aided crowd management systems.
5. The current results set the foundation for more advance work in congestion evaluation applications on real data.
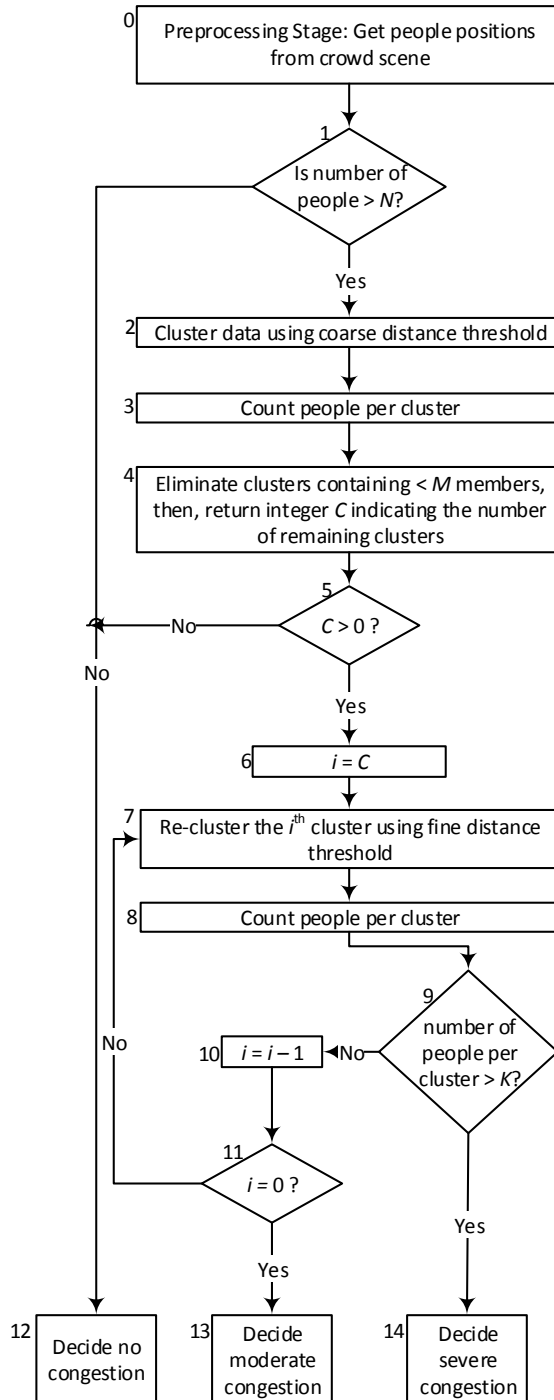
Figures and Tables:

0 — Preprocessing Stage: Get people positions from crowd scene

1 — Is number of people > N?

Yes

2 — Cluster data using coarse distance threshold

3 — Count people per cluster

4 — Eliminate clusters containing < M members, then, return integer C indicating the number of remaining clusters

5 — $C > 0$?

No

No

Yes

6 — $i = C$

7 — Re-cluster the ith cluster using fine distance threshold

8 — Count people per cluster

9 — number of people per cluster > K?

10 — $i = i - 1$

No

No

11 — $i = 0$?

Yes

Yes

12 — Decide no congestion

13 — Decide moderate congestion

14 — Decide severe congestion

Figure 1: Algorithm flowchart

(a)　　　　　　　　　(b)　　　　　　　　　(c)

(d)　　　　　　　　　(e)　　　　　　　　　(f)

(g)　　　　　　　　　(h)　　　　　　　　　(i)

(j)　　　　　　　　　(k)　　　　　　　　　(l)

(m)　　　　　　　　　(n)　　　　　　　　　(o)

(p)　　　　　　　　　(q)　　　　　　　　　(r)
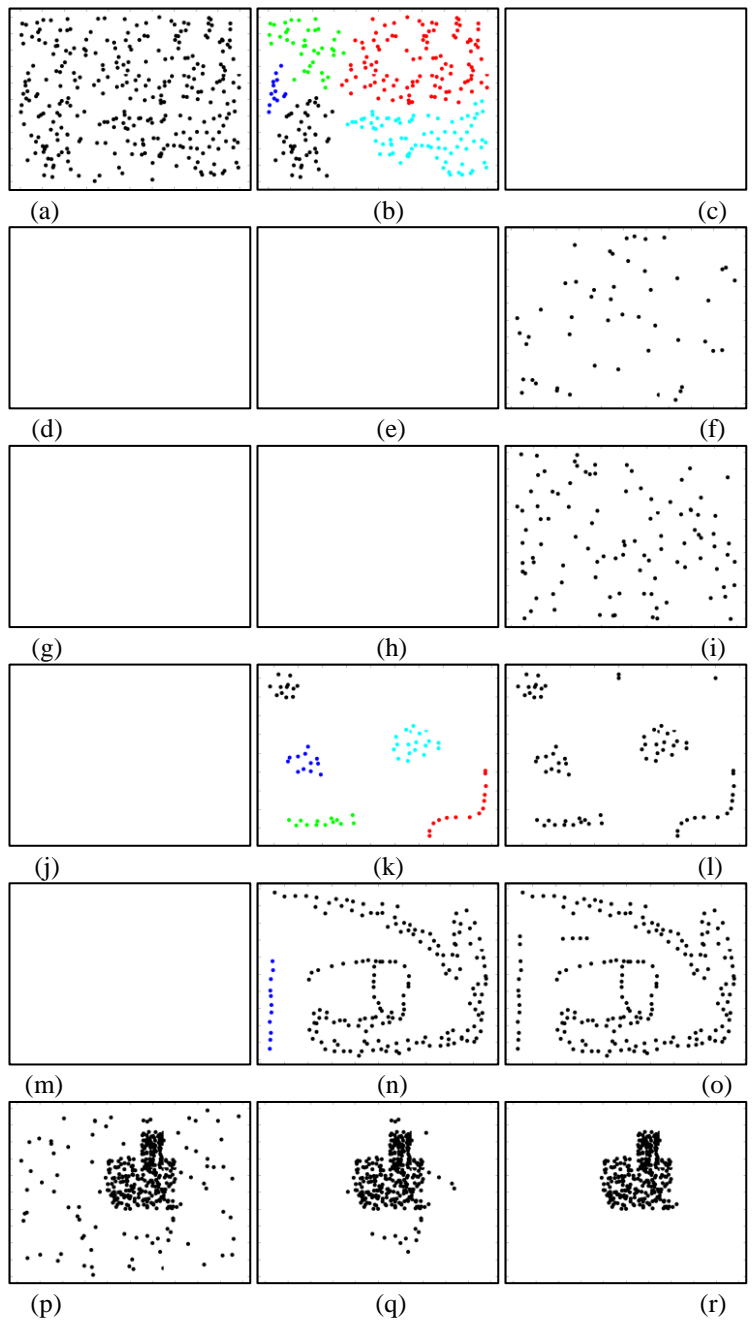
Figure 2: Algorithm test result under different congestion levels

The Custodian of the Two Holy Mosques Institute for Hajj and Umrah Research, Umm Al-Qura University

References:

[1]   M. Gerla and J. Tzu-Chieh Tsai, "Multicluster, mobile, multimedia radio network," Wirel. Networks, vol. 1, no. 3, pp. 255—265, 1995.

[2]   T. F. Gonzalez, "On the computational complexity of clustering and related problems," in System Modeling and Optimization: Proceedings of the 10th IFIP Conference New York City, USA, August 31 -- September 4, 1981, R. F. Drenick and F. Kozin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1982, pp. 174—182.

[3]   H. Steinhaus, "Sur la division des corp materiels en parties," Bull. Acad. Polon. Sci, vol. 1, pp. 801—804, 1956.

[4]   Guojun Shi, Bingkun Gao and Li Zhang, "The optimized K-means algorithms for improving randomly-initialed midpoints," Measurement, Information and Control (ICMIC), 2013 International Conference on, Harbin, 2013, pp. 1212-1216.

[5]   L. Rokach and O. Maimon, "Data Mining and Knowledge Discovery Handbook," O. Maimon and L. Rokach, Eds. Boston, MA: Springer US, 2005, pp. 321—352.