Multimed Tools Appl
https://doi.org/10.1007/s11042-017-5405-3

Localized layout analysis for retargeting of heterogeneous images

Xing Gao[1] **· Guangyu Zhang**[1] **· Juncong Lin**[1] **·**
Minghong Liao[1]

Received: 31 March 2017 / Revised: 13 September 2017 / Accepted: 8 November 2017

Abstract Heterogeneous paper documents (such as newspaper, magazine) are very common in our daily life. They are usually scanned and stored as images. Reading such images on a mobile device is very awkward, as they can only be partially displayed to ensure readability. The user needs to frequently switch among different portions of the image to read clearly. It would be very helpful if the system can automatically determine an appropriate reading area around the user's click position and retarget the area to the whole screen. In this paper, we propose a localized layout analysis method for retargeting of heterogeneous images. Once the user clicks on a fully displayed heterogeneous image, our method can automatically extract an appropriate rectangular region and scale the region to the whole screen for reading. The region is semantically meaningful, and the content is guaranteed to be clear enough when fully displayed on the screen. The experimental results show that our method can effectively avoid those tedious scale and translation operations when reading heterogeneous images, and thus improve the user's experience greatly.

Keywords Heterogeneous image · Localized layout analysis · Image retargeting

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s11042-017-5405-3) contains supplementary material, which is available to authorized users.

✉ Juncong Lin
jclin@xmu.edu.cn

Xing Gao
gaoxing@xmu.edu.cn

Guangyu Zhang
zhangguangyu@stu.xmu.edu.cn

Minghong Liao
liao@xmu.edu.cn

1 Xiamen University, Xiamen, China

Published online: 05 January 2018

1 Introduction

Heterogeneous paper documents (such as newspaper, magazine) are very common. Although more and more documents are produced electronically, people are still accustomed to use paper documents on many occasions especially formal ones. There are many advantages of using paper documents, and most people prefer to read paper documents rather than electronic ones. However, Storing paper documents would be very cumbersome. The ideal solution is to scan and convert paper documents into electronic ones automatically. However, the related techniques (such as optical character recognition) still can not achieve requirements in many aspects including accuracy and efficiency. So most paper documents are simply scanned and stored as images. As a heterogeneous document usually contains various primitives (such as graphs, tables, and textual blocks), the corresponding heterogeneous image needs to have a rather high resolution to ensure clarity.

Many institutes and companies have made great efforts to build digital libraries by scanning paper documents (including books). These efforts greatly help people in acquiring and reading of various documents. With the rapid popularity of mobile devices and enhancement of their performance, people rely more and more on them to read various digital contents. However, these devices usually have a relative small screen. Fully display of a heterogeneous image on these devices would make the content too small to read (Fig. 1a). Thus only a small part of the image can be shown on the screen for reading. And the user needs to frequently switch between zoom in, zoom out and translation to read different parts. Such an experience is apparently awkward and inefficient.

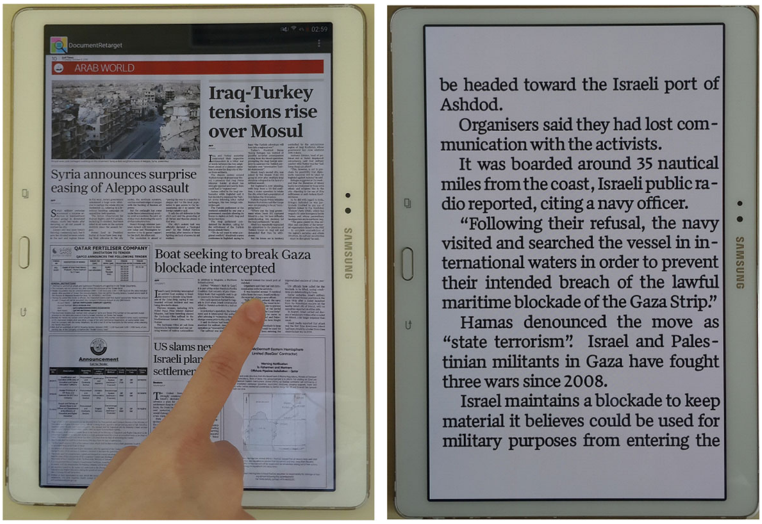
It would be very helpful if the system can automatically determine an appropriate region around the user's click position and retarget the region to the whole screen for reading (Fig. 1b). However, it is not an easy task: (1) the region should be semantically meaningful; (2) the size of the content should be guaranteed to be appropriate when the region is retargeted on the screen; (3) the aspect ratio of the region should approximately match the screen; (4) the algorithm should be fast enough to support fluent reading.

In this paper, we present a novel method for retargeting of heterogeneous images. The user only need to click on the fully displayed image, the system will automatically calculate an appropriate region and retarget it onto the screen. The user can click again to return back to the full-display mode. He can also click on the surrounding area to switch to the neighbor reading region. The main technical contribution is a heuristic method for localized document layout analysis which will identify the homogeneous region around the click position effectively and efficiently.

2 Related works

2.1 Image retargeting

The popularity of mobile devices has greatly stimulated researches in image retargeting. Existing methods can be roughly categorized into discrete (such as cropping and seam carving) and continuous (such as warping) ones. Cropping methods [3, 11, 16, 22] compute an optimal rectangle covering most important areas of the image. However, they may crop out those important regions lying at opposite edges of the image. Seam carving methods [2, 15, 18] resize an image by removing or duplicating seams in less important regions. These methods may suffer from clear artifacts in objects with a well-defined structure.



(a) (b)

Fig. 1 Heterogeneous image retargeting: The user only need to click on the fully displayed image (**a**), our system will automatically calculate an appropriate continuous rectangular area and retarget it onto the screen (**b**)

Non-uniform warping methods formulate the resizing problem as an energy minimization problem which warps a carefully generated rectangular mesh to meet expected aspect ratio. The energy typically measures local deviation of the warp from a shape preserving deformation such as translation [6], rigid transformation [8], or similarity [25, 26]. They can be further classified as linear and non-linear ones. Linear methods are highly efficient as only a sparse linear system need to be solved. Further, factorization of the linear system can be precomputed and reused. However, they usually suffer from artifacts, self-intersetions and foldovers. Non-linear methods can greatly alleviate such artifacts with quite high time cost. Most of them are not suitable for realtime applications. Although some GPU based acceleration methods [9] have been proposed, however, heavy reliance on graphics hardware may hinder applicability to computationally modest platforms such as mobile devices.

Similar to cropping methods, we also compute an optimal rectangle which will be retargetd to the whole screen. However, the definition of optimal rectangle varies. We emphasis on the semantical meaning of the region and the guarantee clarity other than reservation of salient objects.

2.2 Document layout analysis

Document layout analysis tries to decompose a document image into many different regions such as texts, images, separators, and tables. Many approaches have been proposed, which can be categorized into four main types: bottom-up, top-down, hybrid, and multi-scale resolution method. Bottom-up methods [1, 14, 20] start with local information such as words and finally merge them into blocks or paragraphs. They are usually applicable to various layouts. But their complexities are generally quadratic in time and space, or even more. Top-down methods [7, 23] look for global information on the entire document page and split it into blocks and finally into words. These methods usually involve linear complexity and efficiency when documents have Manhattan layout, however, the paper documents today are varied and plentiful. Hybrid methods [4, 21, 24] focus on the analysis of the connected components and the white spaces between them. However, the results obtained are not convincing, especially with the classification of non-text regions. Multi-scale resolution methods [5, 10] yield positive results by using multi-scale resolution. However, the computation is complex and requires a quite long time. The main differences between our work and these methods lie in: (1) We perform layout analysis locally (around the click position) other than globally (the whole document), leading to a fast and accurate layout analysis method; (2)The size of content should be guaranteed when the area is retarget onto the screen; (3)Block merge and split are involved to construct region with appropriate aspect ratio.

There are also some works on synthesizing and editing of document layout. In [12], Peter ODonovan et al described an energy-based model for evaluating layouts based on graphic design principles and stylistic goals. The model was demonstrated to be used in layout synthesis, layout retargeting and layout improvement. They also introduced a system to help novice users creating graphic design by making interactive layout suggestions [13]. Our localized layout analysis can also be used to edit the layout of existing document as demonstrated in Section 7.2.

3 Pilot study

Before designing the algorithm, we conducted a pilot experiment to validate the necessity of our technique and to determine optimal values of some parameters. 20 participants were recruited in the study, they were asked to finish several tasks as described below.

In the first task, we would like to investigate how long can a participant bear to watch a document once it has been loaded (named load delay d_l) and how long can he wait before switching to a different portion(named switch delay d_s). if d_l is large enough, then we can simply perform global layout extraction on loading and quickly determine the most appropriate reading region. While d_s put a requirement on the performance of localized method. Once the participant loaded a document, a timer was set to let him wait for a random integer number of seconds, where the number is between 0 and 5. He was then asked to rate whether the waiting time is acceptable. Similar instructions were given to rate the switch delay. Each participant were asked to rate three different d_l and d_s respectively. Each time value was assured to be rated 10 times.

The purpose of the second task is to determine the comfortable character size for reading. We prepared several images with the same resolution. Each of them contains a homogeneous textual region with the same font size. For each participant, we randomly chose an image and scaled it to let the averaged diagonal length of characters' bounding boxes to be τ mm. τ is randomly set to be an integer between 1 and 10. The image is shown to the participant with the scale. And the participant is asked to rate the preference of the character size. The process will be repeated 4 times for each participant. Each value of τ is assured to be selected 8 times.

The purpose of the final task is to observe when would the participant switch to different portions other than reading the whole image in the screen. We synthesized 10 heterogeneous images with different size. When they are fully displayed, the character size will be exactly from 1 mm to 10mm. Each participant was randomly shown four different images in full-display mode and asked to read through each of them. We then asked each of them to rate the reading experience when the image are initially shown to them. We also recorded the number of participants who scaled the document when reading. Each image was assured to be shown to 8 different participants.

From results of the pilot study, we can draw the following conclusions:

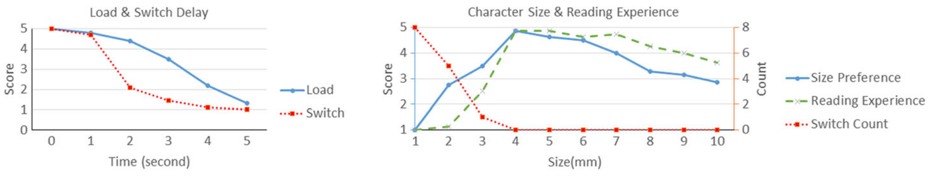
1. Localized layout analysis is necessary. We can see from the blue curve in Fig. 2a that the score drop below neural (3) when $d_l \geq 4$. Thus it is infeasible to simply extract global layout and use it to retarget the image, as the process usually requires more than 4 seconds (See Table 3 in Section 7). Similarly, it is better to keep the time cost of localized layout analysis less than 2 seconds according to the red curve in Fig. 2a.
2. Acceptable character size is greater than 3 mm (according to the blue curve in Fig. 2b). This is an important suggestion for determine the dimension of retargeting region in localized layout analysis.
3. Although some participants can bare to read the content with a small character size without any switch operations (red curve in Fig. 2b), their experience drop quickly (green curve in Fig. 2b). Thus quick switch between different portions is expected providing the user interface is friendly.

4 User interface

Figure 3 shows the interface of our system. The user clicks on any position in the fully displayed document (Fig. 3b), our algorithm will automatically extract the rectangle of homogeneous region around the clicked position(red rectangle in Fig. 3a). The rectangle may be split or extended so that its aspect ratio can roughly match the screen aspect ratio. The rectangle area is then resized to fit the screen for reading. The user can click around the center to return back to the full-display mode. He can also switch to neighboring region by clicking around the surrounding area (Fig. 3c, d, e).

5 Localized layout analysis

A naive method to determine the appropriate reading region is to globally extract the layout of whole image and then merge or split homogeneous regions. However, as pointed out in Section 3, it will cause large delay on loading and thus affect users' experience. Besides, due to the complexity of the heterogeneous image, layout extracted with global methods usually



(a) (b)

Fig. 2 Pilot Study: **a** How the delay on loading and switching will affect participants' experience; **b** Participants' preference on character size and how it will affect their reading experience

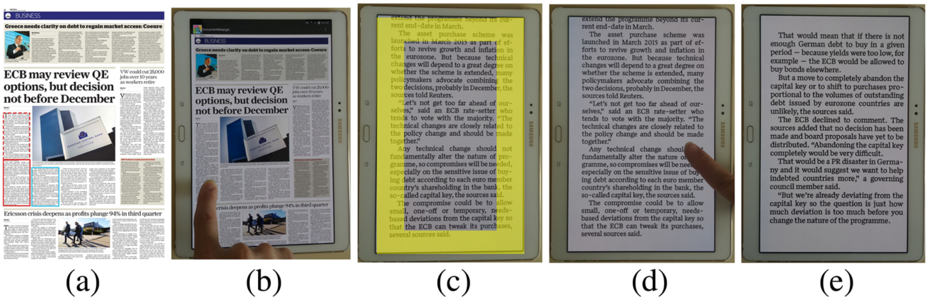


Fig. 3 The interface of our system: **a** Heterogeneous image with the calculated retargeting area (red rectangle) around the user's click position and its neighboring area (blue rectangle); **b** The user clicks on the document under full-display mode; **c** The screen is divided into surrounding area (yellow zone) and center area in the retarget mode, with different response to the user's click; **d** The user clicks on the surrounding area of retargeted region; **e** Switch to neighboring region

contains a lot of small fragments. These fragments present great challenges for the following operations such as merge, split and retargeting. We notice that the user usually clicks inside the reading area. Based on such an observation, we design a progressive local layout analysis method which expands horizontally and vertically from the clicked position. Information obtained during the process will be used to guide following expansion steps. The flowchart of the method is demonstrated in Fig. 4. Given a heterogeneous image, we firstly convert the image into binary form with the method proposed by Faisal Shafait [17]. A rectangular area $O_s(W_s, H_s)$ centering at the clicked position $\mathbf{p}(x, y)$ is determined with the initial width W_s and height H_s set as 50 px. R_s will be expanded at a ratio of γ_s (γ_s=1.5) until it contains some non-empty pixels. We then extract and analysis connected components from these non-empty pixels. We take different strategies according to the different categories of seed components:

– image component: calculate the bounding rectangle;
– table component: calculate the bounding frame;
– textual component: go through a iterative region growth to extract homogeneous region;

We finally retarget the bounded region of interest to the whole screen for reading.

5.1 Classification of connected component

We treat differently on different connected components: those considered as noise will be discarded; image components can be directly retargeted to the screen; and for textual components, we do layout analysis to extract the homogeneous region for retargeting. We investigate the following characteristics to decide the category of a connected component CC_k:

– $B(CC_k)$: bounding box of CC_k, where $\mathbf{p}_B^l(x_B^l, y_B^l)$ is the left up corner and $\mathbf{p}_B^r(x_B^r, y_B^r)$ is the right bottom corner.
– A_k: the area of CC_k, defined as number of pixels contained in CC_k.
– $A_k^B = W_k \times H_k$: the area of $B(CC_k)$, where W_k is the width of $B(CC_k)$, H_k is the height of $B(CC_k)$, W_k and H_k are both measured in pixels.
– $D_k = A_k / A_k^B$: density of CC_k.

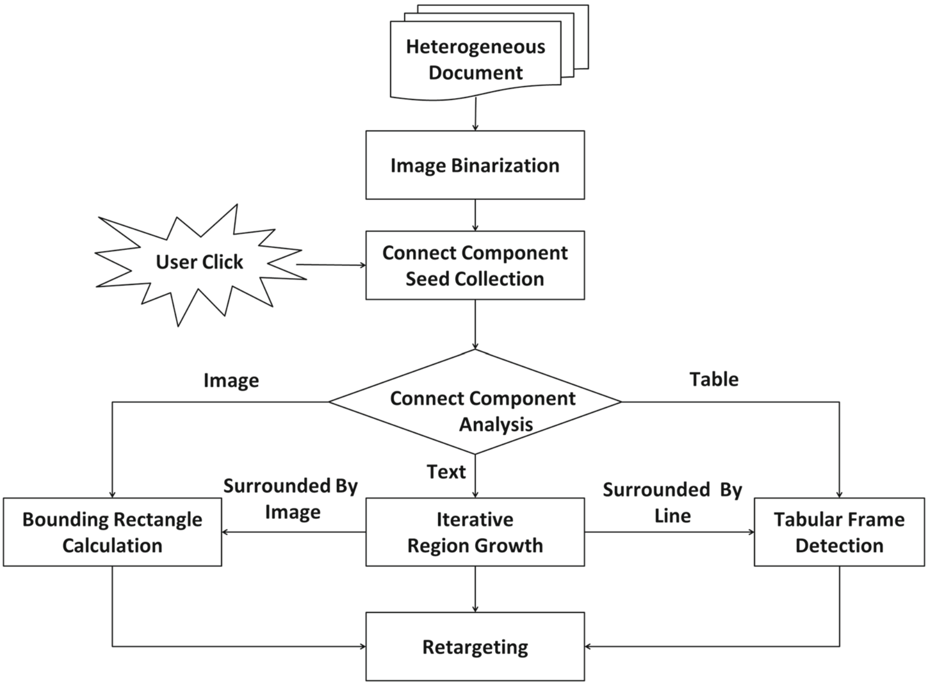


Fig. 4 Flowchart of the localized layout analysis method

- N_k: the number of other connect components inside $B(CC_k)$.
- $T_k = min(W_k, H_k)/max(W_k, H_k)$: aspect ratio of $B(CC_k)$.

A connected component CC_k is treated as noise when its area A_k is smaller than a given threshold A^t, $A^t = 6$ px in our implementation.

We consider a connected component CC_k as a non-textual component when the following conditions are satisfied:

- $N_k > N^t$, $B(CC_k)$ contains more than N^t other connected components, $N^t = 3$.
- $D_k < D^t_l$, the density is smaller than a given threshold, $D^t_l = 0.1$.
- $D_k > D^t_u$, the density is larger than a given threshold, $D^t_u = 0.9$.

5.2 Iterative homogeneous region growth

We notice that textual components usually obey the mahantton layout except those around image components. We thus design a region growing method which exploits the inherent constraints among rows of textual components: (1) Expand horizontally from seed components to generate seed rows which will then be put in a queue Q in a top-to-down order. For each row R_k, we store its two end positions $\mathbf{p}^l_k$ and $\mathbf{p}^r_k$, two boundary tags F^l_k and F^r_k; (2) Update each row in the queue mutually according to manhatton constraints; (3) Pick the first/last row from the queue and expand up/down to get seed textual components for new rows. These steps are repeated until the terminal condition is satisfied. The pseudo code is shown in Algorithm 1. Below, we explain some details about the algorithm.

Algorithm 1 Iterative homogeneous region growth

Require: A binarized heterogeneous document I, Set of seed textual components SC found around the user's click positions

Ensure: Bounding rectangle R of the homogeneous region around the user's click position.

```
int vertical_expansion = 0
Queue Q
Set SR
while vertical_expansion! = 3 do
    for each seed textual component CC_i ∈ SC do
        if the row of CC_i is in the queue then
            continue
        end if
        Expand horizontally from CC_i to get seed row R_i
        put R_i into SR
    end for
    clear the set SC
    for each row R_i ∈ SR do
        put R_i into Q
        mutual update rows in Q
    end for
    for each row R_i ∈ SR do
        Expand vertically to get new seed textural component NCC_i
        put NCC_i into SC
    end for
    if vertical expansion then
        set vertical_expansion = 3
    end if
end while
Calculate the bounding rectangle R of rows in Q
```

Horizontal expansion To expand the row from a seed component CC_k, we find a new connected component CC_n in the rectangular area D_r next to the right side (or left side) of CC_k. The dimension of the rectangle is set to be $(W_r = \lambda_h \cdot W_k, H_r = H_k)$. We set the row boundary flag $F_k^r = 1$ (or $F_k^l = 1$) if no CC_n is found. Otherwise, we check CC_n with the homogeneity criteria. We take CC_n as the new seed and repeated the process if it passes the check. And we set $F_k^r = 2$ (or $F_k^l = 2$), if it can not pass. $\lambda_h = 2$ in our experiments.

Mutual row updating Once a new row R_n is added into the queue Q, we will perform the mutual row updating process. We compare the end positions $\mathbf{p}_n^l$ and $\mathbf{p}_n^r$ of row R_n with those of each other row R_k, and adopt different actions according to the comparison (taking the right side for example):

– $(x_n^r > x_k^r + \lambda_m \cap (F_n^r \neq 2)$: expand row R_n in the right side until $(F_n^r \neq 0) \cap (x_n^r > x_k^r - \lambda_m)$.
– $(x_k^r < x_n^r - \lambda_m \cap (F_n^r \neq 2)$: expand row R_k in the right side until $(F_k^r \neq 0) \cap (x_k^r > x_n^r - \lambda_m)$.

We do nothing for other cases. Once the comparison has finished, we choose the row with maximum x^r (minimum x^l for the left side case) and repeat the above comparison process. The whole process will be repeated until no expansions happen. $\lambda_m = 3$ px in our implementation.

Vertical expansion The basic idea of vertical expansion is similar to horizontal expansion. Given the top (or bottom) row R_i of current region, a seed component CC_i in R_i is chosen and an initial rectangle D_t is constructed above (or below) CC_i with $(W_t = W_i, H_t = \lambda_h \cdot H_i)$. We then find the seed component CC_n in the next row in D_t. However, as shown in Fig. 5, we may not be able to find CC_n if the next row is an ending row or there is a gap in the row. To address this problem, we choose several positions (left, right and the middle) from R_i to expand. $\lambda_h = 1.5$ in our implementation.

Homogeneity criteria Our homogeneity criteria considers both the property of connected components and the property of the gap between neighboring components. A connected component CC_k can be added to the current expanding row only when the area of its bounding box A_k^B satisfies:

$$| A_k^B - A_{avg}^B | \leq A_{HC}^B. \tag{1}$$

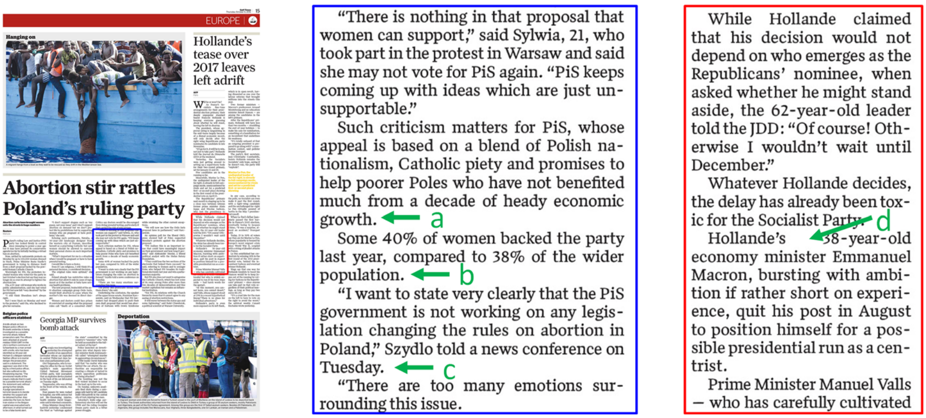
And the gap width G_k^w between the bounding box of CC_k and the bounding box of the neighboring connected component in the row satisfies:

$$| G_k^w - G_{avg}^w | \leq G_{HC}^w. \tag{2}$$

For vertical expansion, we have a similar homogeneous constraint on gap height G_k^h between neighboring rows:

$$| G_k^h - G_{avg}^h | \leq G_{HC}^h. \tag{3}$$

Here, A_{avg}^B is the averaged area of all the connected components in current expanding row, A_{HC}^B is the homogeneity threshold of connected component, we set $A_{HC}^B = \lambda_c \cdot A_{avg}^B$, G_{HC}^w and G_{HC}^h are homogeneity thresholds for gap width and height respectively. We set $G_{HC}^w = \lambda_{gw} \cdot G_{avg}^w$ and $G_{HC}^h = \lambda_{gh} \cdot G_{avg}^h$. G_{avg}^w is the averaged gap width for an expanding row while G_{avg}^h is the averaged gap height among all neighboring rows in queue Q. $\lambda_c = \lambda_{gw} = \lambda_{gh} = 0.5$ in our experiments.



"There is nothing in that proposal that women can support," said Sylwia, 21, who took part in the protest in Warsaw and said she may not vote for PiS again. "PiS keeps coming up with ideas which are just unsupportable."
Such criticism matters for PiS, whose appeal is based on a blend of Polish nationalism, Catholic piety and promises to help poorer Poles who have not benefited much from a decade of heady economic growth.
Some 40% of women backed the party last year, compared to 38% of the wider population.
"I want to state very clearly that the PiS government is not working on any legislation changing the rules on abortion in Poland," Szydlo told a news conference on Tuesday.
"There are too many emotions surrounding this issue.

While Hollande claimed that his decision would not depend on who emerges as the Republicans' nominee, when asked whether he might stand aside, the 62-year-old leader told the JDD: "Of course! Otherwise I wouldn't wait until December."
Whatever Hollande decides, the delay has already been toxic for the Socialist Party.
Hollande's 38-year-old economy minister Emmanuel Macron, bursting with ambition if rather short on experience, quit his post in August to position himself for a possible presidential run as a centrist.
Prime Minister Manuel Valls – who has carefully cultivated

Fig. 5 Singular cases that may lead to failure in finding seed component: **a**, **b**, **c** ending row; **d** gap in the row

5.3 Table detection

Table detection is a challenging problem as there are various types of tables. To simplify the problem, we limit our table detection to those have clear lines. They can be classified into three categories: *closed table* with full horizontal and vertical connections; *line table* with only horizontal or vertical lines; *non-close table* lacking one or more horizontal and/or vertical components. For closed tables and non-close tables, we can extract the connected component CC_t of all the lines as they are connected together. We can then consider CC_t as a table if it satisfies the following condition:

- The density of CC_t is less than D_l^t.
- $B(CC_t)$ contains many text elements.

For *line table*, the key is to correctly detect lines. Once a horizontal/vertical line is detected, we can than expand vertically/horizontally as we do in homogeneous region growth. We notice that line elements always have great disparity in height and width. And their density is very high. So we can identify a line component CC_l when the density $D_l \geq D_u^t$ and the aspect ratio $T_l \leq 0.1$.

6 Homogeneous region merge and split

The extracted homogeneous region may be still too large to display on the whole screen, making it difficult for the user to read the content clearly. It may be too small, leading to extraordinary large texts which is also not comfortable for the user. To address these problems, we need to further merge or split homogeneous regions. The key here is to assure the size of the textual components appropriate when the region is retargeted on to the screen.

We have already known the optimal character size $S_c = 3$mm in Section 3, defining by the diagonal length of character's boundingbox. We choose to merge homogeneous regions when the averaged screen size of the textual components $\overline{S}_t$ in the retargeted region satisfies: $\overline{S}_t > \tau_u \cdot S_c$. Similarly, we choose to split a homogeneous region when $\overline{S}_t < \tau_l \cdot S_c$. The merge (or split) direction is chosen to make the aspect ratio of the retarget region as close as possible to that of the screen. $\tau_u = 1.5$ and $\tau_l = 0.8$ in our implementation.

7 Experimental results

We have implemented a prototype with Java on Android platform (SAMSUNG Galaxy Tab S T800, quad core CPU@1.3GHZ+1.9GHZ, 3GB RAM, aspect ratio 1.6). Figures 6, 7 and 8 show retargeting results of textural area, image and table on mobile devices respectively. We also compare the localized layout analysis algorithm with existing global layout extraction methods [21] in Fig. 9. We can clearly see from the figure that the global method tends to generated fractals while our method can assure a continuous appropriate region. We also perform a similar comparison with Chinese documents. And similar conclusions can be drawn from Fig. 10. Parameters for the two figures are summarized in Table 1. Figure 11 shows how different settings of user-tuned parameters can affect the localized layout analysis. Roughly speaking, strict constraints on homogeneity may cause various break on expanding.

Fig. 6 Retargeting results of textural area

Fig. 7 Retargeting results of image area

Statistics of results in Figs. 6, 7 and 8 are listed in Table 2. It takes about 1 second for our system to retarget in all cases, which meets the requirement of an interactive application quite well. For comparison purpose, we also count the time of global layout extraction for these documents with [21]. The timing statistics are listed in Table 3.

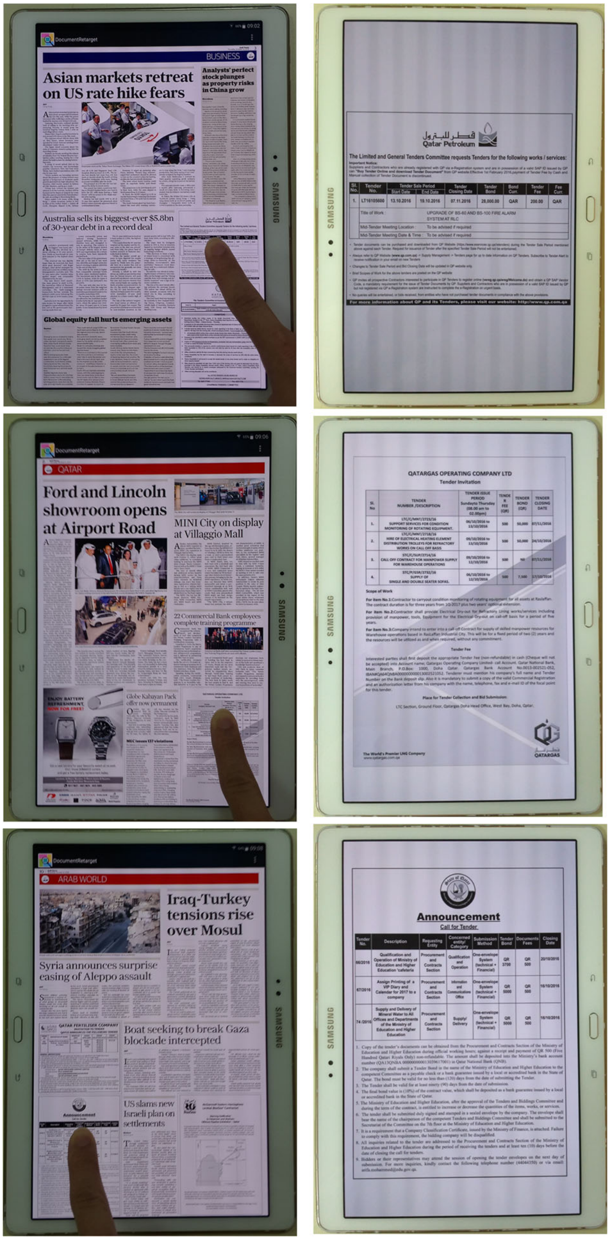


Fig. 8 Retargeting results of table area

7.1 User study

We conducted user studies to validate that our system can greatly help the user reading heterogeneous image. 20 participants were recruited, 14 males and 6 females aged from 20 to 24. They were undergraduate or post-graduate students from computer science. The study

Fig. 9 The comparison between our localized layout analysis algorithm and the existing global one [21]: the left column shows the local layout extracted on the clicked position (red circle) with our method, while the right column shows the global layout extraction with [21]

Fig. 10 The comparison between our localized layout analysis algorithm and the existing global one [21] with Chinese documents: the left column shows the local layout extracted on the clicked position (red circle) with our method, while the right column shows the global layout extraction with [21]

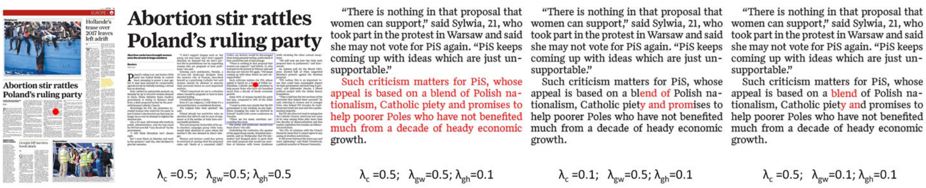
Table 1 Parameter setting for different language

Parameter	Description	Value	
		English	Chinese
D_l^t	Lower threshold for component density	0.1	0.1
D_u^t	Upper threshold for component density	0.9	0.9
λ_h	Step size for horizontal expansion	2	2
λ_v	Step size for vertical expansion	1.5	1.5
λ_m	Threshold for mutual row updating	3	3
λ_c	Threshold for component homogeneity	0.5	0.3
λ_{gw}	Threshold for horizontal gap width homogeneity	0.5	0.3
λ_{gh}	Threshold for vertical gap height homogeneity	0.8	0.3
τ_l^t	Threshold for region split	0.8	0.8
τ_u^t	Threshold for region merge	1.5	1.5

consisted of three stages: The first stage investigated the random access ability of the system; The second stage emphasized on quick location of certain interested area on the image; The third stage evaluated how the system can help to improve users' reading experience and efficiency on mobile devices. At the beginning of the user study, we introduced our system and tutored each participant. Each participant was then given 2 minutes to get familiar with the system. The whole process took about 5 minutes. To compensate potential discrepancy about familiarity with the interfaces, participants were equally divided into two groups: G1: 8 males, 2 females, mean age 22; and G2: 6 males, 4 females, mean age 22. Both groups were asked to perform the same tasks but with different orders. Participants of G1 were asked to do the job with our system first, while those in G2 were asked to do without our system first.

Experiment 1 In this experiment, participants were asked to count the number of a given word (”to”) in a heterogeneous image with and without our system. We record the time when a participant was conducting the task. The statistics are given in Fig. 12b which depicts clearly that our tool can save bout 65% of the time.

Experiment 2 Participants were required to locate the position of a randomly specified paragraph and display it appropriately for reading with and without the retargeting tool. From the time statistics in Fig. 12c, we can see that our tool can save about 10% time.



(a) (b) (c) (d) (e)

Fig. 11 Effect of user-tuned parameters: **a** heterogeneous image and clicked position; **b** result generated by default setting; **c–e** results generated with varied parameters (Red characters are included in the interested region)

Table 2 Timing statistics of the heterogeneous image retargeting method

Figure		Properties of retargeting region			Cost(s)
	length-width ratio	number of CCs	total area of CCs		
	R1C2	1.91	124095	649	0.644
	R1C3	1.59	118060	708	0.691
Fig. 6	R2C2	1.61	193312	505	0.504
	R2C3	1.14	158477	355	0.445
	R3C2	1.64	117950	543	0.542
	R3C3	1.42	94302	483	0.482
Fig. 7	R1C2	0.63	559149	1	0.474
	R2C2	0.97	2068139	1	0.821
	R3C2	1.44	1740629	75	0.759
Fig. 8	R1C2	0.26	133959	305	0.393
	R2C2	0.51	399681	422	0.741
	R3C2	0.52	240842	679	1.11

Note: RiCj means the subfigure on the i-th row j-th column of the corresponding figure.

Experiment 3 We let each participant read through a given heterogeneous image with and without our system for both groups. The timing statistics given in Fig. 12d shows that our tool can save 4% time.

Generally, our system can greatly avoid the frequent switch of scale and translation as we can see from the statistics of Experiment 1. It may seems not so apparent from the statistics of experiment 2 and 3 for real reading tasks. This is because most of time is spend on reading. Actually, we can clearly find that our method can save around 10 seconds during the two experiments for each user.

Rating Once a participant finished the three experiments, he was asked to rate the system on aspects including “easy to learn”, “easy to use”, “helpful”, “preference” with a scale of 1 to 5 (1 means “strong disagree” and 5 means ‘strongly agree’). We performed Wilcoxon signed rank tests [19] on the questionnaire data to evaluate the significant effects of the availability of our systems on the listed aspects, with the result shown in Fig. 13.

Table 3 Timing statistics of the global layout extraction method

Figure		Cost of global layout analyze (s)
Fig. 6	R1C1	6.818
	R2C1	15.701
	R3C1	18.686
Fig. 7	R1C1	18.502
	R2C1	19.343
	R3C1	7.039
Fig. 8	R1C1	19.504
	R2C1	17.828
	R3C1	18.625

Note: RiCj means the subfigure on the i-th row j-th column of the corresponding figure.

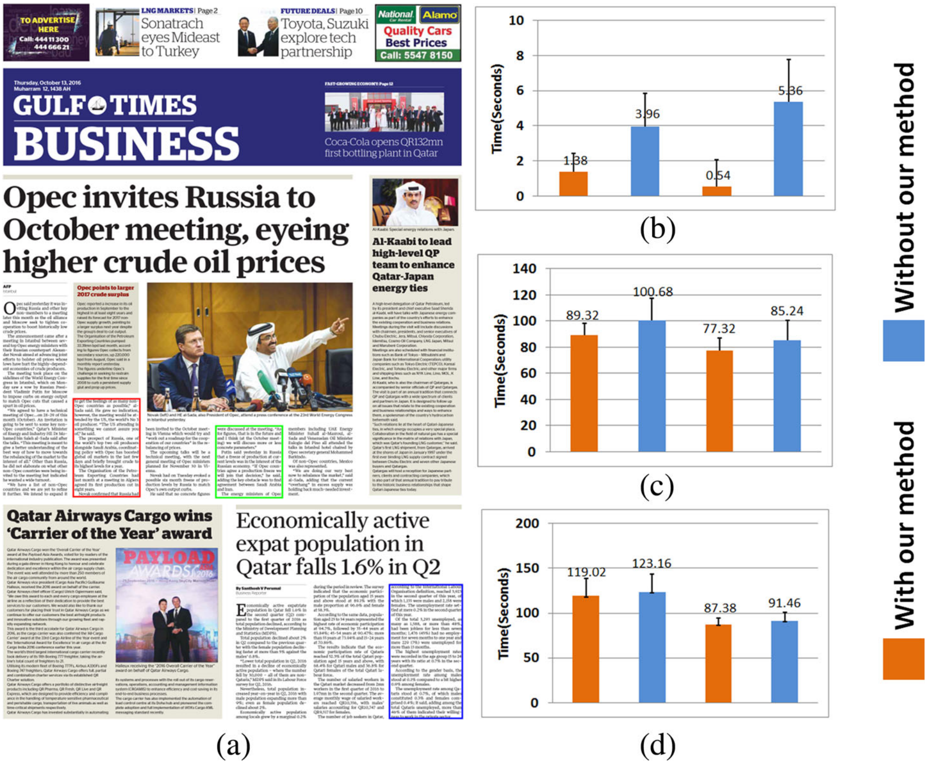


Fig. 12 User study: **a** Heterogeneous image used in experiments; **b** Timing data for experiment 1; **c** Timing data for experiment 2; **d** Timing data for experiment 3;

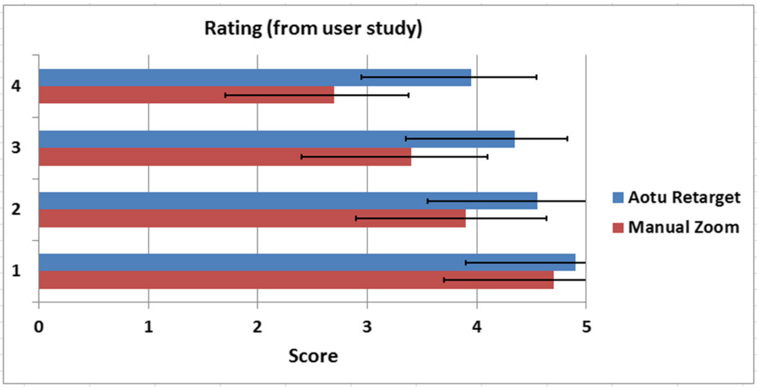


Fig. 13 Users' rating of our system on a scale between 1 and 5. (1 = strongly disagree, 5 = strongly agree), the error bars show the 95% confidence intervals

In summary, our system can greatly save the time cost for users in general reading activities. The system is helpful and quite preferred by most of the participants.

7.2 Applications

Thanks to the localized layout analysis, we can edit a heterogeneous image in a higher level way. For example, we can rearrange paragraphs (or tables, images etc), change their appearance (such as color), scale or delete them very convenient. Figure 14 shows some examples.

(c)

Fig. 14 Other applications of the localized layout analysis algorithm: **a** Rearrange paragraphs; **b** Change appearance; **c** Construct a new heterogeneous image by synthesizing primitives from existing ones

8 Conclusions

Heterogeneous images are very common. Due to their high resolution, reading such images on mobile devices is extremely uncomfortable. In this paper, we present a heterogeneous image retargeting method to improve the reading experience of these images on mobile devices. Once the user clicks on the screen, our system will perform a localized layout analysis to quickly extract a rectangular area on the image which is semantically meaningful, with appropriate aspect ratio and scale. The area is then resized to the whole screen for reading. Once the user finishes reading, he can click around the center to return back to the whole image or click around the surrounding area to switch to next area. Our system can greatly relief the user from frequent scale and translation of the image, thus greatly improve the efficiency and experience.

8.1 Limitations

The main problem with our method is that it currently relies on many user-tuned parameters. However, we find these parameters mainly depend on the language during our experiments. Thus we do not need to frequently adjust them for images with the same language. Another possible limitation maybe the assumption on manhatton layout of the input image. Our localized layout analysis exploits the constraints between rows in manhatton layout. The algorithm may fail when encountering compact-arranged non-manhatton layout. However, we barely could find such documents during our experiments.

References

1. Agrawal M, Doermann D (2009) Voronoi++ A dynamic page segmentation approach based on Voronoi and Docstrum features. In: Proceedings of 10th international conference on document analysis and recognition, pp 1011–1015
2. Avidan S, Shamir A (2007) Seam carving for content-aware image resizing. ACM Trans Graph 26(3):Article No. 10
3. Chen LQ, Xie X, Fan X, Ma WY, Zhang HJ, Zhou HQ (2003) A visual attention model for adapting images on small displays. Multimed Syst 9(4):353–364
4. Chen K, Yin F, Liu CL (2013) Hybrid page segmentation with efficient whitespace rectangles extraction and grouping. In: Proceedings of 12th international conference on document analysis and recognition, pp 958–962
5. Cheng H, Bouman CA (2001) Multi-scale bayesian segmentation using a trainable context model. IEEE Trans Image Process 10(4):511–525
6. Gal R, Sorkine O, Cohen-Or D (2006) Feature-aware texturing
7. Jaekyu Ha, Haralick RM, Phillips IT (1995) Recursive X-Y cut using bounding boxes of connected components. In: Proceedings of 3th international conference on document analysis and recognition, pp 952–955
8. Karni Z, Freedman D, Gotsman C (2009) Energy based image deformation
9. Krähenbühl P, Lang M, Hornung A, Gross M (2009) A system for retargeting of streaming video. ACM Trans Graph 28(5):Article No. 126
10. Lee SW, Ryu DS (2001) Parameter-free geometric document layout analysis. IEEE Trans Pattern Anal Mach Intell 23(11):1240–1256
11. Liu F, Gleicher M (2005) Automatic image retargeting with fisheye-view warping
12. O'Donovan P, Agarwala A, Hertzmann A (2014) Learning layouts for single-page graphic designs. IEEE Trans Vis Comput Graph 20(8):1200–1213
13. O'Donovan P, Agarwala A, Hertzmann A (2015) Designscape: design with interactive layout suggestions. In: Proceedings of the 33rd annual ACM conference on human factors in computing systems, pp 1221–1224

14. O'Gorman L (1993) The document spectrum for page layout analysis. IEEE Trans Pattern Anal Mach Intell 15(11):1162–1173
15. Rubinstein M, Shamir A, Avidan S (2008) Improved seam carving for video retargeting. ACM Trans Graph 27(3):Article No. 16
16. Santella A, Agrawala M, DeCarlo D, Salesin D, Cohen M (2006) Gaze-based interaction for semi-automatic photo cropping
17. Sauvola J, Pietikäinen M (2000) Adaptive document image binarization. Pattern Recogn 33(2):225–236
18. Shamir A, Avidan S (2009) Seam carving for media retargeting. Commun ACM 52(1):77–85
19. Siegel S (1956) Non-parametric statistics for the behavioral sciences. McGraw-Hill, New York
20. Simon A, Pret JC, Johnson AP (1997) A fast algorithm for bottom-up document layout analysis. IEEE Trans Pattern Anal Mach Intell 19(3):273–277
21. Smith R (2009) Hybrid page layout analysis via tab-stop detection. In: Proceedings of 10th international conference on document analysis and recognition, pp 241–245
22. Suh B, Ling H, Bederso BB, Jacobs DW (2003) Automatic thumbnail cropping and its effectiveness
23. Sun H-M (2005) Page segmentation for Manhattan and non-Manhattan layout documents via selective CRLA. In: Proceedings of 8th international conference on document analysis and recognition, pp 116–120
24. Tran TA, Na IS, Kim SH (2016) Page segmentation using minimum homogeneity algorithm and adaptive mathematical morphology. Int J Doc Anal Recogn 1–19
25. Wang YS, Tai CL, Sorkine O, Lee TY (2008) Optimized scale-and-stretch for image resizing. ACM Trans Graph 27(5):Article No. 118
26. Zhang GX, Cheng MM, Hu SM, Martin RR (2009) A shape-preserving approach to image resizing. Comput Graph Forum 28(7):1897–1906

Xing Gao is an associate professor at the Software School of Xiamen University, P.R.China. He received the Ph.D. degree in Computer Science from Harbin Institute of Technology. His research interests include Computer Graphics, Image Processing.

Guangyu Zhang is a MS student at the Software School of Xiamen University, P.R.China. His research interests include Image Retargeting and User Interaction.

Juncong Lin is an associate professor at the Software School of Xiamen University, P.R.China. He received the Ph.D. degree in Computer Science from Zhejiang University. His research interests include Geometry Processing and Image Authoring.

Minghong Liao is a professor of Software School of Xiamen University, P.R.China. He received his Ph.D. degree in Computer Science in 1993 from Harbin Institute of Technology of China. His research interests include Intelligent Network, Pervasive Computing and Computer Graphics.