

---

This is an electronic reprint of the original article.  
This reprint may differ from the original in pagination and typographic detail.

Montoya Freire, Maria L.; Potts, Dominic; Dayama, Niraj Ramesh; Oulasvirta, Antti; Di Francesco, Mario

## Foraging-based optimization of pervasive displays

*Published in:*  
Pervasive and Mobile Computing

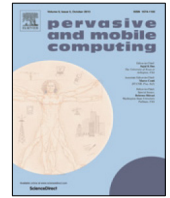
*DOI:*  
[10.1016/j.pmcj.2019.02.008](https://doi.org/10.1016/j.pmcj.2019.02.008)

Published: 01/04/2019

*Document Version*  
Publisher's PDF, also known as Version of record

*Published under the following license:*  
CC BY-NC-ND

*Please cite the original version:*  
Montoya Freire, M. L., Potts, D., Dayama, N. R., Oulasvirta, A., & Di Francesco, M. (2019). Foraging-based optimization of pervasive displays. *Pervasive and Mobile Computing*, 55, 45-58.  
<https://doi.org/10.1016/j.pmcj.2019.02.008>



# Foraging-based optimization of pervasive displays

Maria L. Montoya Freire<sup>a,\*</sup>, Dominic Potts<sup>c</sup>, Niraj Ramesh Dayama<sup>b</sup>,  
Antti Oulasvirta<sup>b</sup>, Mario Di Francesco<sup>a</sup>

<sup>a</sup> Department of Computer Science, Aalto University, 02150, Finland

<sup>b</sup> Department of Communications and Networking, Aalto University, 02150, Finland

<sup>c</sup> Department of Computing and Communications, Lancaster University, LA1 4WA, United Kingdom



## ARTICLE INFO

### Article history:

Received 22 December 2018

Received in revised form 25 February 2019

Accepted 26 February 2019

Available online 2 March 2019

### Keywords:

Information foraging theory

Computational design

Display foraging

Pervasive displays

User study

Field study

## ABSTRACT

The article addresses a key challenge in the design of content for pervasive displays: how to engage passers-by who have limited time and attention? To achieve this, we apply a novel approach for computational design of interesting display content using tiled layouts. We present a model of *display foraging* based on information foraging theory to describe the behavior of a rational but time-limited user looking at a display. Accordingly, our work aims to maximize the information gain for tiled displays. This complex problem is divided into two phases: (1) generating designs of tiled layouts and (2) assigning content options to individual tiles based on what predicted by display foraging. Accordingly, a proof-of-concept system was realized then evaluated computationally and empirically with a control study and field study. The results show that the proposed system can engage significantly more people than typical digital signage.

© 2019 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

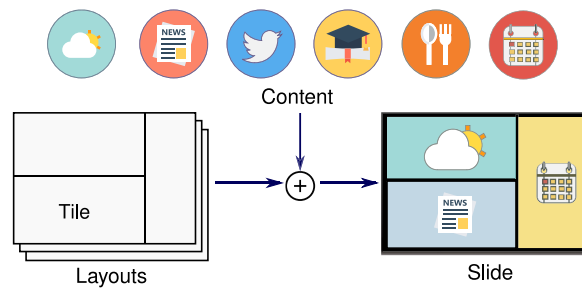
## 1. Introduction

Pervasive displays are increasingly employed in public spaces such as airports, train stations, universities, and malls. Several solutions for public spaces have been proposed in the state-of-the-art including e-Campus [1], UBI-hotspots [2], and ReflectiveSigns [3]. In general, displays convey contextual information to passers-by, allowing them to become aware of relevant matters in a short time [4]. However, engaging users is a key design goal that remains challenging to reach. People have limited time and attention; consequently, they ignore displays that they deem neither informative nor interesting [5]. Traditional solutions for digital signage organize content as a slideshow, wherein contents are displayed one slide at a time with a set time period [6]. A different approach is tiling: a display is partitioned into several tiles, each showing its own content [7]. Squeezing in as many tiles as possible is still not a feasible option, though, since it may cause information overload for users [8]. Therefore, it is an important research problem to determine the criteria for selecting and showing content on a display, based on time-dependent user interest [9].

Most of existing works targeted user engagement by dynamic content adaptation [10] or by allowing explicit user interaction in several ways [11]. Examples include employing large multi-touch screens [12] or standard displays with input peripherals such as push buttons [13], cameras [3], and eye-trackers [14,15]. Mobile devices have also been proposed, for instance, to support user profiles [16–18] or as a means to control the content shown by displays [6,11,19].

\* Corresponding author.

E-mail addresses: [maria.montoyafreire@aalto.fi](mailto:maria.montoyafreire@aalto.fi) (M.L. Montoya Freire), [d.potts2@lancaster.ac.uk](mailto:d.potts2@lancaster.ac.uk) (D. Potts), [niraj.dayama@aalto.fi](mailto:niraj.dayama@aalto.fi) (N.R. Dayama), [antti.oulasvirta@aalto.fi](mailto:antti.oulasvirta@aalto.fi) (A. Oulasvirta), [mario.di.francesco@aalto.fi](mailto:mario.di.francesco@aalto.fi) (M. Di Francesco).



**Fig. 1.** The display foraging model derives the time a user spends by looking at a display, given several types of content and possible arrangement of tiles (i.e., layouts). An optimization process obtains the actual content for the display (i.e., a slide), by selecting content options and assigning them to tiles, so as to maximize the perceived user interest.

Similarly, user representations along with proxemic parameters (e.g., distance, orientation) have been employed to entice user interaction [20,21]. Other works, instead, analyzed the impact of external factors on user engagement with displays, for instance, the visibility of a display in a given location as well as the audience behavior in front of the display [22,23].

We complement this body of work with a solution that can be applied to pervasive displays, either with or without touch capabilities. In particular, we focus on non-interactive displays showing contextual information to people passing by, which are prevalent in current deployments [8]. This article addresses the creation of interesting content for pervasive displays as a computational design problem [24]. Specifically, we aim to maximize the *information gain* of a tiled display, a quantitative description of the benefit a user receives from looking at the display. We target users who are *rational* (i.e., utility-maximizing) but *busy* (i.e., time-limited), as it is the case in real deployments [25]. To do so, we take an approach that builds on *information foraging theory* (IFT), which is a quantitative model of a rational but time-limited agent seeking and consuming information, similar to how animals forage for food [26]. On these grounds, we propose a model of *display foraging* to describe the time a user spends on looking at a display as an estimation of their interest [3,13]. In contrast, IFT has been previously applied, among others, to understand navigation behavior in interactive information hierarchies [27], program code [28,29], Web [30], visualization dashboards [31], focus+context visualizations [32], and image search [33]. To our knowledge, information search interfaces is the only application area where interface contents have been programmatically adapted [34]. For instance, Goodwin et al. [35] leveraged IFT to search a medical database and determine which concepts users should attend to. Indeed, we are not aware of IFT applied to pervasive display scenarios.

Display foraging provides the grounds to generate content, however, its arrangement onto a tiled display is a challenging problem per se [36]. Due to this, we decompose the computational problem of maximizing the information gain of tiled displays into two phases. Firstly, we generate a set of candidate layouts as collection of tiles that fulfill the design criteria of both efficiency (the screen estate is fully utilized) and compatibility (tiles are arranged meaningfully). This approach to tiled display uses ideas from layout partitioning using constraint-based methods [37] and grid layout generation [38]. Secondly, we assign specific among numerous content options to tiles by evaluating the total gain of a given layout, according to what predicted by display foraging. Also, we devise a proof-of-concept system for non-interactive displays, then, we evaluate such a system both computationally and empirically. For the latter, we carry out a controlled user study to assess the accuracy of display foraging, and a field study to demonstrate the effectiveness of the system in the real-world. The results show that our system is practical and significantly improves user engagement compared to a typical solution for digital signage.

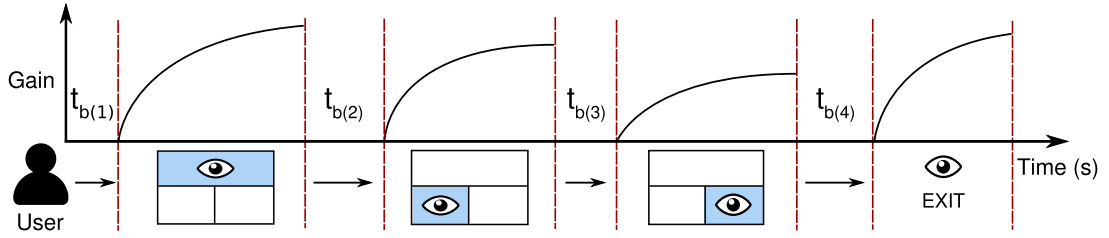
## 2. Modeling display foraging

This section introduces the proposed display foraging model. We begin with the relevant background and an overview of the information foraging theory behind the model.

### 2.1. Background

As shown in Fig. 1, we refer to content shown at a certain time on a display as a *slide*. Each slide contains multiple types of content at the same time, with individual items obtained from a data source such as a weather API or RSS feeds. For instance, a display can present on the same slide the weather forecast for the current week and the upcoming events in a certain place. The portion of the slide associated with specific content is called a *tile*. The arrangement of the individual tiles on a slide is called a *layout*.

A key assumption in the display foraging model is a rational user – acting according to preferences or goals – who decides which tiles to attend to and how long to do so before moving on to do something else. This principle is used for estimating the best possible information gain on a given layout. For the definitions, we build on information foraging theory [39]. Foraging theories draw a parallel between how animals forage for food and people navigate information



**Fig. 2.** The display foraging model determines which tiles a rational user would visit, in which order. The bottom portion shows the sequence of tiles perused by the user in a layout. At the top are the gain functions for the content in the respective tiles and the time needed to move from one tile to another.

spaces. The environment of an agent contains patches separated by a given distance (or, equivalently, a certain travel time). A patch is associated with a *gain*: a function describing how quickly energy or information is obtained within that patch. Animals choose patches for maximizing the expected energy obtained per unit of time. For instance, a wolf can choose to hunt for rabbits or instead deer: the optimal choice is governed by expected energy-intake gain per unit of time. Rabbits are easier prey, but deer offer larger energy gains. Similarly, information foragers (rational users) maximize the expected information gain per unit of time as a tradeoff between interest in (or importance of) the content and the effort (or time) to extract the information from it [40].

There are two major models in IFT work [26]. In the *patch model*, the environment is modeled as a set of patches, each having a specific gain function. The question that arises is how long an agent should stay in a patch before moving on to the next one. The optimal time for leaving a patch is when the net gain does not increase any further, since the informative content has been exhausted. In the *diet model*, there are different types of patches, characterized by a certain profitability. The question here involves determining which patches to visit and in which order so as to maximize the overall gain before leaving the environment. Patches are ranked by profitability and sorted in such a way that prioritizing in order of decreasing profitability is considered the optimal sequence for visiting the patches. The agent stops foraging once the cumulative rate of gain is greater than the profitability of the next patch.

In our formulation, we combine the two models and aim to derive the *within-patch* and *between-patch* behavior for a given pervasive display layout. In other words, we seek to uncover the order of visiting the patches (tiles) and how long the user stays in each to maximize the information gain.

## 2.2. Foraging behavior

The display foraging model defines how interesting certain content must be for getting displayed on a specific tile. Within the IFT framework, each tile is a patch<sup>1</sup> with an associated *gain* – the value acquired after foraging a tile. At a given time, the user may be visiting a specific tile or may be moving from one to the next. In particular, the user spends time  $t_w$  *within* a tile by consuming its content; then, the user takes time  $t_b$  to move *between* the two tiles. A key assumption in our model is that a user can always choose *not* to attend to the display: we model such behavior by adding an *exit tile*, which does not correspond to any “physical” tile. The exit tile has its own information gain function. The gain of such a tile is high for a busy user; in contrast, it is much lower for a user who is, for instance, sitting in a waiting room. The relative gain of the exit tile compared to the others ultimately determines which tiles go in a display layout.

Fig. 2 illustrates our IFT-based model applied to a layout containing three tiles. The upper part of the figure shows the gain functions corresponding to the individual tiles. The flat portions of the curves indicate the time between tiles, while those with non-zero values represent logarithmic gain functions whose parameters describe individual tiles. Note, that each tile has its own logarithmic gain function, generally different from others. The lower part of the figure depicts the foraging process of the user as the sequence of perused tiles. Specifically, the user starts looking at the tile at the top in the layout, moves to the bottom-left one, and then looks at the one in the lower right. Finally, the user stops paying attention to the content on the display, moving to the exit tile. In the example shown, the user reaches the exit tile after scanning all tiles in the layout. This does not necessarily happen in general – the user may decide to stop looking at the display at any time, on the basis of the cumulative gain of the tiles traversed as compared to that associated with the exit tile.

As noted above, IFT uses a logarithmic function to model the information gain within a tile [40], defined as

$$g(t) = a \cdot \ln(t) - b \quad (1)$$

Here, the gain over time is affected by the parameters  $a$  and  $b$ : the slope of the related function and its offset on the vertical axis, respectively. These parameters can be obtained by applying curve fitting to an empirical distribution for a given tile. We discuss a method used to achieve this in Section 6.1. But how long does a user stay in a tile? In other words,

<sup>1</sup> For clarity, we will simply use the word “tile” in the rest of the article.

how can we derive within-tile time  $t_w$  for a given gain function? Even though the user obtains some gain in a foraged tile, that gain does not increase any further (or increases only very slowly) after a certain time. Hence, a rational user moves from the current tile to the next one at that exact time. Indeed, the optimal time to stop foraging in a certain tile can be derived through the marginal value theorem [41] and the *rate* of gain:

$$R = \frac{g(t_w)}{t_b + t_w} \quad (2)$$

In particular, the optimal time  $t^*$  to stay in a tile is until the instantaneous increase in the gain equals the rate of gain:

$$g'(t^*) = R \quad (3)$$

where  $g'(t^*)$  is the first-order derivative of the gain function  $g(t)$  evaluated at time  $t^*$ . Solving the equality yields

$$\frac{a \cdot \ln(t^* - t_b) - b}{t^*} - \frac{a}{t^* - t_b} = 0 \quad (4)$$

which determines the optimal time  $t_w = t^*$  within the tile.

### 2.3. Obtaining parameter values

Below, we introduce a formulation specifically tailored for pervasive displays to determine the IFT parameters.

#### 2.3.1. Between-tile parameters

Recall that time between tiles  $t_b$  is the time expended in moving from one tile to another. This can be derived on the basis of a characterization of human gaze [42]. In particular, we model the time between tiles as the sum of two components: saccade duration  $t_f$  (namely, the time for the eyes to *find* a piece of content — specifically, that displayed in a tile) and time  $t_r$  taken to *recognize* a certain type of content. We employ Baloh et al.'s characterization of saccade duration [42] to derive  $t_f$  and set  $t_r$  to 400 ms per Chen et al. [43], thus obtaining

$$t_b = t_f + t_r = 37 + 2.7 \cdot \alpha + 400 \quad (5)$$

where  $\alpha$  is the angular distance between two tiles (expressed in degrees). In practice, we compute this by taking the Euclidean distance between the midpoint coordinates of individual tiles and converting it to an angle on the basis of the display's size and resolution. This yields an optimistic estimate of  $t_b$ , under the assumption that the user can directly fixate at the target. That is not a problem, though, since the validity of the *absolute* prediction does not matter for the optimization process, whereas the *relative* time predicted between two foraged tiles correlates with true search times.

Note that the time between tiles can be fully described from the given layout, independently of the content. In other words, for a given layout, we can calculate the time between every two tiles from their geometry and their position on the screen. Indeed, the method described above can model a user looking at tiles in arbitrary reading order, as long as the corresponding sequence is known. Hence, this method also works for natural reading order (i.e., left–right, top–bottom).

#### 2.3.2. Within-tile parameters

Recall that time within tiles  $t_w$  is determined by the gain function in Eq. (1), specifically by the parameters  $a$  and  $b$ . Because the value of  $t_w$  signifies the time needed by the user to consume the tile's content, it can be described through two components. The first is related to the parameter  $a$  and is determined by its specific content, in terms of the related interest; we model this through the coefficient  $i$ . The second is related to the geometry of the tile — to its size  $s$  (normalized with respect to the screen size) and aspect ratio  $r$  (i.e., the ratio between its width and height). We can express the following relationship accordingly:

$$\frac{a}{i} + \frac{b \cdot s}{r} = t_t \quad (6)$$

This gives an indication of the time a user is willing to spend looking at a certain tile. The rationale behind the equation is that a user is willing to take longer if the content is deemed of interest, and the time needed to read that content is proportional to the size of the corresponding tile, weighted by its aspect ratio. Of the coefficients we have just introduced, only interest,  $i$ , is unknown; those related to the geometry of the tile can be derived from the layout.

Now we need to introduce another relation to determine the  $b$  coefficient. To this end, we leverage the optimal foraging time from Eq. (4), thereby obtaining

$$b = \frac{i \cdot r \cdot t_t \cdot (t_t \cdot \ln(t_t) - (t_b + t_t))}{i \cdot s \cdot (t_t \cdot \ln(t_t) - (t_t + t_b)) + r \cdot t_t} \quad (7)$$

where  $t_b$  is the time between the tile considered and the subsequent one (derived as described earlier).

It is worth noting that the parameters associated with the gain function can be derived in several ways, with one example being a user study as detailed in Section 6.1.

### 3. Optimization approach

The computational problem of generating tiled content is split into two parts here: (1) choosing a layout that meets the given design constraints and (2) assigning content to tiles.

#### 3.1. Generation of candidate layouts

While numerous options are available for arranging tiles into a layout [37,38], we are interested in solutions that satisfy the following design criteria: the tiles should not overlap, should not overflow the viewport of the display (i.e., they should not be cropped), and there should be no unused space between tiles. The first two constraints ensure that the individual tiles are entirely visible on the display, and the third guarantees that the solution obtained is space-filling — that is, the entire area of the display is used. In addition to these design constraints, we consider tile-specific restrictions in terms of minimum (maximum) size and aspect ratio.

The literature presents several techniques for optimal partitioning of a target area. Among them, geometric programming [44] has been leveraged for floor planning to arrange rooms with given sizes into the area with the minimum bounding box. Also, several types of cutting problems have been formulated with the goal of minimizing wasted material, in contexts such as paper rolls [45]. Regrettably, these approaches are not able to guarantee satisfying all the requirements listed above. While some specialized solutions do exist that fulfill them [36], these are extremely expensive from the computational perspective. For instance, traditional approaches to preventing overlap consider the individual pixels of a display: if no pixel is assigned to two or more tiles, there is no overlap [46,47]. Though it is rather straightforward, this approach requires a huge number of variables — e.g.,  $2 \cdot 10^6$  for a display with a resolution of 1,920 by 1,080 pixels — so involves a very long execution time.

To address this issue, we introduce an approach based on integer linear programming (ILP). Our solution is able to generate various layouts that comply with the design criteria<sup>2</sup> stated above and also with the tile-specific geometry constraints. In particular, the layout generator produces a set of feasible solutions whose maximum size can be explicitly set. For instance, the generator can consider the minimum width, height, and aspect ratio for five tiles and generate up to eight layouts satisfying these constraints together with the aforementioned design objectives.

The ILP formulation requires decision variables that specify the position of all tiles (i.e., their coordinates and sizes). The conditions of no overflows and no unused space can simply be enforced in terms of the absolute positions and sizes of all tiles. In contrast, the no-overlap constraint cannot be easily described with these variables. To address this issue, decision variables are employed to express a tile's position with respect to others. For instance,

$$L_{ij} = \begin{cases} 1 & \text{if tile } i \text{ is to the left of tile } j \\ 0 & \text{otherwise} \end{cases}$$

is one of these decision variables. Similarly,  $R_{ij}$ ,  $T_{ij}$ , and  $B_{ij}$  describe, respectively, whether tile  $i$  is to the right of, above, or below tile  $j$ . The no-overlap constraint is then enforced by the following condition:

$$1 \leq L_{ij} + R_{ij} + T_{ij} + B_{ij} \leq 2$$

#### 3.2. Assignment of contents to tiles

After generating layouts, we apply the display forager model as the objective function through which the optimizer maximizes the information gain. For this, we require a layout and a list of content options with their corresponding gain parameters. Recall that a layout is simply a collection of tiles on the screen; as such, it only represents a way to partition the screen into tiles. To describe the actual partition, a layout contains information about the size of the tiles and their locations on the display.

In general, the number of content options is higher than the number of tiles, making the selection a challenging problem. We address this challenge with an exhaustive method that considers all feasible option combinations to determine the best content assignment for a given layout, within a given time budget.

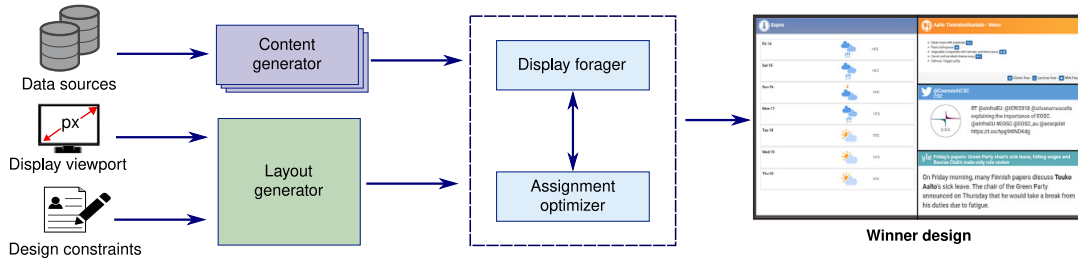
Content density is not an issue, since the content adapts to the size and aspect ratio of the tiles. Thus, the criteria for selecting the content options from those available are based on their respective gains under the display foraging model.

### 4. System design and implementation

Below, we introduce the components of our IFT-based system to optimize display content, followed by the details of a prototype implementation and its performance.

<sup>2</sup> We intentionally do not define a utility function for layouts, but only constraints, so as to use an optimizer as a generator of candidate layouts.





**Fig. 3.** IFT-based optimization system. Content is generated from data sources, while candidate layouts are determined on the basis of the physical properties of the display along with the design constraints on a meaningful arrangement of tiles. The optimizer evaluates the layouts by assigning content to tiles and obtaining their gain in accordance with the display foraging model. The best layout is the one with the highest gain, which becomes the winning design shown on the display.

#### 4.1. System overview

Fig. 3 illustrates the components of our system. One of the key elements is a set of *data sources* providing information in a certain domain. A *content generator* takes the data source as input and generates the content that is shown in a tile. For illustrative purposes, let us consider the case of weather forecasts. In this case, the data sources are weather data provided by, for instance, a meteorological institute or a third-party provider. They might show a city's daily forecast for the next three days. The other core element is the actual *pervasive display*, which is characterized by some physical operation parameters: resolution (in pixels), physical size, and location. In addition, a specification of *design constraints* is available to describe what is a meaningful arrangement of tiles, from the designer perspective. For example, the tiles may have a minimum and maximum width or height, and there may be permissible ranges for their aspect ratios also. A *layout generator* takes the physical information for the display and the design constraints as input. The result is a collection of layouts – arrangements of tiles – that satisfy the design constraints and are fit for that specific display.

The *display forager* employs the IFT-based model in Section 2 to characterize the time a user is looking at the display. Specifically, the display forager takes a layout as input, calculates the time between tiles, and then evaluates the total gain achieved with different content choices. The *assignment optimizer* evaluates several candidate layouts by deriving the highest total gain achieved by the possible content-option combinations. In doing so, the optimizer also considers the case in which the user stops looking at the display after consuming the content in a subset of the tiles in a layout (i.e., the user attending to the exit tile). The final result of the optimization is the best assignment of content to tiles in terms of the total information gain. Once such an assignment is found, the tiles in the chosen layout are filled with content and shown as a slide, as illustrated at the far right of Fig. 3.

#### 4.2. Prototype implementation

We implemented a prototype of our system according to the client–server model: the pervasive display runs a client application that receives the content from a server in the cloud [48,49]. The server performs the most computationally expensive tasks: layout generation and content assignment.

We implemented a web-based application based on HTML5 and the Bootstrap toolkit [50]. On the server side, we employed Node.js [51] along with the Express framework. A layout, alongside its assigned content, is sent to the client (a web browser) through message sockets offered by the Socket.io library [52]. On the client side, the web application utilizes a *div* element that represents a tile and jQuery for handling the content to be shown therein. Several *divs* are initially created without any style, just to populate the screen in line with the winner design. Individual styles are then created for each *div* to show the content, depending on the assigned tile. The content is responsive, in the sense that it automatically adapts to the size and aspect ratio of the tiles. Fig. 4 shows three sample layouts generated by the system according to the scenarios presented in the next section.

The layout generator instantiates the problem formulated in Section 3 with IBM ILOG CPLEX (version 12.6.3). We precompute the layouts for the viewport size of the display. The server runs the display forager and the assignment optimizer on the generated layouts to compute the best possible slide. The assignment optimizer was implemented in python and applied a random multi-start method [53] to find the best assignment of contents to tiles. Such methods employ multiple starting points to explore the search space, thereby increasing the chances of finding a global optimum.

### 5. Walk-through

This section shows how our system can be applied in various scenarios and evaluates the performance of the optimizer.

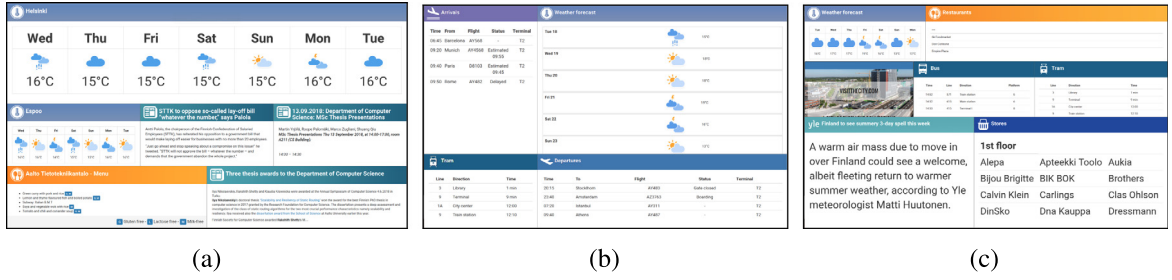


Fig. 4. Layouts obtained from the optimizer for the scenarios considered: the (a) university, (b) airport, and (c) mall.

### 5.1. Scenario-specific characterization

We devised three distinct illustrative scenarios, to help us evaluate the display foraging model for several places and contexts.

#### 5.1.1. The university

In the first, there is a pervasive display in a break area at our university. This is an open space where people take breaks or gather for various purposes: having a snack or even lunch, drinking coffee or tea, speaking with colleagues, reading research papers, etc. This area is open to anyone — the staff, students, and visitors alike. Among the data sources for the application running for the display are upcoming events in the building, including seminars, workshops, and defense of doctoral dissertations; news from national agencies and social media (e.g., tweets); the current weather and the forecast for the rest of the week; the menus of the university restaurants and nearby cafes; and the schedules for the closest public-transportation connections.

#### 5.1.2. The airport

Here, a pervasive display is placed in the public-access area of an international airport. This is where family members and friends await arrivals, and all passengers arrive here once they have collected their baggage, before proceeding to their final destination. The space is a gateway to transportation by taxi, train, or bus, and it leads also to a tourist information point; a small supermarket; and several cafes. The data sources in this context are weather information and forecasts; the schedule and status of incoming flights; advertisements for local events and entertainment; tourist information; and directions to transportation.

#### 5.1.3. The shopping mall

The final scenario's pervasive display is in the main lobby of a shopping mall, a large, several-stores building. People enter the lobby from an underground parking area or a nearby bus stop. There are several types of customers: shoppers visiting the mall for the first time, in search of a specific store; casual visitors who may be hungry or simply passing by; and regular customers arriving for daily grocery shopping. The data sources for this specific place are the stores' locations and opening hours; current store-specific sales and promotions; local events, including those at the public library next door; the bus times for the closest stop; and menus for outlets in the food court.

### 5.2. Performance evaluation

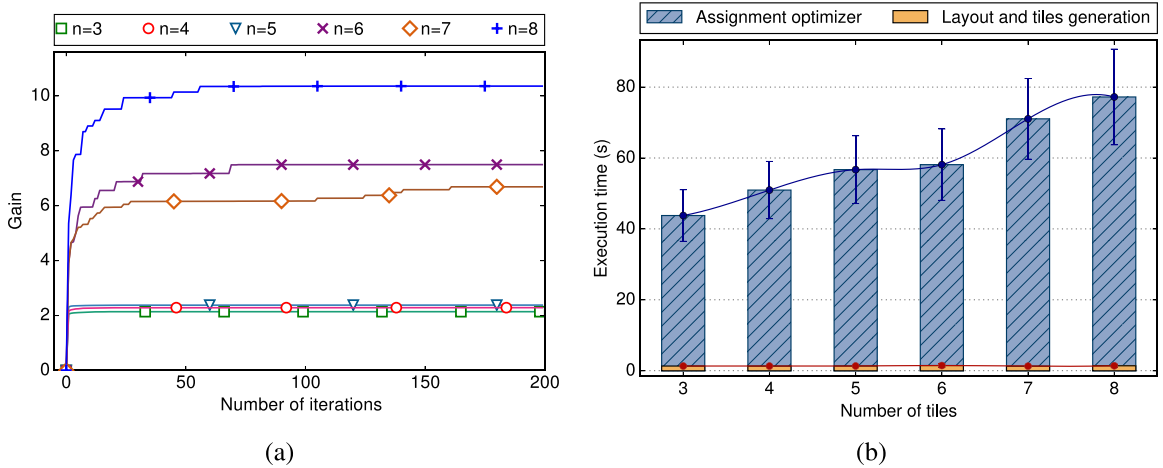
To evaluate the efficiency of the optimizer for the prototype implementation presented in Section 4, we considered the data sources described above for the university scenario, with the physical parameters as described further on.

We characterize our system by two key metrics: *convergence* toward the chosen slide, in terms of information gain as a function of the iterations by the optimizer, and the *execution* time, the time taken by the system to generate the best possible slide. For both, we consider layouts with 3–8 tiles and report the average values obtained by running each experiment five times. Our system ran on a virtual machine with 15.6 GB of RAM, 6 VCPUs, and 80 GB of storage.

#### 5.2.1. Convergence

Fig. 5(a) illustrates the convergence of the optimizer. In the experiments, we set the maximum number of iterations to 200 and used 31 distinct layouts. The results shown are grouped by number of tiles  $n$  in the source layout, independent of their actual arrangement. The figure clearly shows that the information gain always reaches convergence in the considered scenario. Convergence is fast, requiring fewer than 10 iterations, when the number of tiles ranges from three to five. Layouts with a higher number of tiles (i.e., 6–8) need more time to converge, but the required number of iterations remains well below the maximum in the experiments. That said, the actual speed of convergence varies between the layouts with the three highest gains. In particular, a larger number of tiles does not necessarily imply a higher gain: layouts with seven tiles result in a lower gain than those with six. This is because feasible layouts depend on size and aspect ratios of tiles. Consequently, layouts with many tiles may result in rather odd arrangements due to geometry constraints, thereby affecting the resulting gain.





**Fig. 5.** Performance evaluation for our system prototype: (a) convergence of the total gain produced by the optimizer and (b) a breakdown of the execution time by component, as a function of the number of tiles in a layout, for 200 iterations.

### 5.2.2. Execution time

Fig. 5(b) illustrates the execution time of the system, broken down by component: layout and tile generation as well as the assignment optimizer. The figure also shows the standard deviation in the (average) results with error bars for the case of 200 iterations. The figure clearly shows the time needed for layout and tile generation to be roughly constant and equal to a few seconds (less than 1.5 s), almost irrespective of the number of tiles in a layout. The assignment optimizer, instead, takes significantly more time to derive the winner design. This occurs since the number of data sources available (15) is much higher than the maximum number of tiles in the layouts considered. Also, execution time clearly increases with the number of tiles, from 43 s (for layouts with three tiles) to 78 (for layouts with eight tiles), because more assignments must be evaluated by the assignment optimizer. The results reported here show that it can take up to one and a half minutes to obtain the best possible slide. However, it is important to note that the optimization can be executed in advance; for slowly changing content, a set of designs in a slideshow can still be precomputed. Also note that our proof-of-concept implementation could be made faster by parallelizing the optimizer and running it on multiple servers in the cloud.

## 6. Controlled evaluation

This section presents a user study with gaze tracking, conducted to evaluate the accuracy of the proposed system: how closely the display foraging model can predict users' behavior in the real-world. Before that, we introduce some groundwork, a survey-based calibration study to derive the IFT parameters in the display foraging model.

### 6.1. Estimation of gain parameters through survey data

Previous work has developed techniques for estimating users' interest in pervasive displays and their content [54,55]. Alike, we designed a survey-based study for rapid collection of input that could be used to seed real deployments. Only two dimensions were used: interest and time spent. As we show later, the quality of inputs collected in this manner sufficed.

In preparation for the study, we derived a set of 28 layouts produced by our system. Specifically, we used a set of 15 distinct content options based on the data sources defined for the university scenario (as we target participants at a university) and assigned them to the individual tiles in the layouts considered. Then, we selected specific tiles on the basis of their diversity in geometry (size and aspect ratio) and their frequency in the set of considered layouts. The goal was to single out different tiles with the same content to include in the survey.

The study used an online survey with 60 question items, showing 15 tiles by using four distinct tile sizes. These values were selected to reduce the time needed to complete the full survey to a maximum of 30 min. Each tile was shown (separately from the others) for eight seconds, after which a popup form appeared with the following items<sup>3</sup>:

**Q1:** I find the shown content *interesting*.

**Q2:** I would spend over 15 seconds<sup>4</sup> to examine this content.

<sup>3</sup> Italicized text was emphasized in the survey.

<sup>4</sup> We verified that the distribution of responses was statistically significant at the end of the user study, thereby supporting such a setting for the threshold.

Participants could select values on a 5-point Likert scale from “Strongly disagree” (1) to “Strongly agree” (5). The recruitment process was done through email and social networks, and movie tickets were raffled off as an incentive for participation. All respondents volunteered under informed consent. We collected data from 99 participants (71 male and 28 female), aged between 18 and 54 years. Most subjects (90%) were working or studying at our university, so they were familiar with the content used in the survey. The other 10% were former students and visitors.

We used the values collected from the survey (see also [Appendix](#)) to derive the gain functions as follows. We employed question Q1 to directly quantify the coefficient  $i$  in Eq. (6). In particular, we calculated the average value reported across all participants for that question. We employed question Q2, in turn, to derive the time the user would spend on looking at a certain tile. Specifically, we normalized the average value of the survey results and then weighted it with a reference time window. In detail, the time is

$$t_t = t_{min} + \frac{\bar{q} - q_{min}}{q_{max} - q_{min}} \cdot (t_{max} - t_{min}) \quad (8)$$

where  $\bar{q}$  is the average of the question values;  $q_{min}$  and  $q_{max}$  are the minimum and maximum value reported for the question, respectively; and  $t_{min}$  and  $t_{max}$ , respectively, are the minimum and maximum amount of time a user was willing to look at the tile. We set  $t_{min} = 8$  as the time the tile was shown in the survey, while we set  $t_{max} = 23$  so as to have a reference time window centered at 15 s, which is the specific value referred to in the question. After obtaining the  $i$  and  $t_t$  coefficients for each tile, we solved the system of equations given by Eq. (6) and Eq. (7) by adding the quantities related to the size and aspect ratios so as to derive the  $a$  and  $b$  coefficients of the gain functions in Eq. (1).

Since not all geometries were associated with tiles used in the survey, we had to estimate the value of  $q$  for all sizes and aspect ratios that could be generated by the system. In doing so, we first calculated the two parameters

$$q_w = \frac{1}{n} \sum_{n=1}^4 \bar{q}_n \cdot \frac{w_{max} - w_{min}}{w_{max} - w_{min} + |w - w_n|} \quad (9)$$

$$q_h = \frac{1}{n} \sum_{n=1}^4 \bar{q}_n \cdot \frac{h_{max} - h_{min}}{h_{max} - h_{min} + |h - h_n|} \quad (10)$$

for a tile of size  $w$  by  $h$  pixels. The value for  $q_w$  was computed as the sum over the four tile sizes, considering only their width  $w_n$  and the rating  $\bar{q}_n$  obtained in the user study. The computation also requires to determine  $w_{max}$  and  $w_{min}$ , obtained by calculating the range where a tile fits when placed alongside the four tiles. The same rationale applies for computing  $q_h$  but considering only the height of the tiles. Finally, we calculated  $\bar{q}$  for each tile as the geometric average of  $q_w$  and  $q_h$ , namely,  $\bar{q} = \sqrt{q_w \cdot q_h}$ .

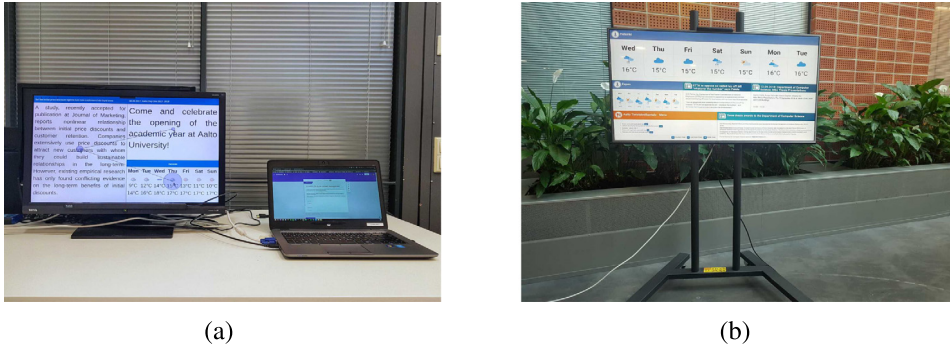
## 6.2. Method: gaze-tracking-based user study

We now turn to the controlled user study to validate our proposed display foraging model and evaluate its accuracy. This user study applied gaze tracking. Specifically, we used gaze measurements to obtain the total foraging time as the time the user spent perusing the tiles shown on the display.

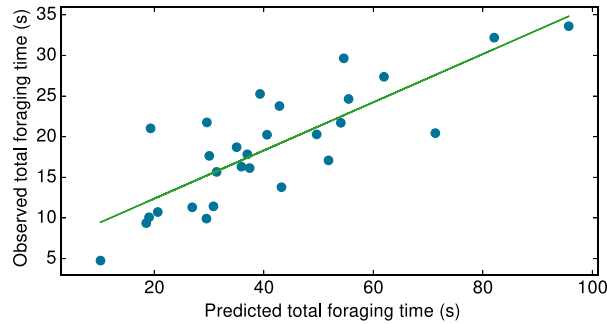
To conduct this study, we recruited volunteers<sup>5</sup> who work at our university. In total, 30 people took part in the study, 20 male and 10 female; 67% of the participants were aged 25–34, 30% were between 18 and 24, and 3% were of age 35–44. To record gaze data, we employed a Tobii Eye Tracker 4C mounted on a 24” display with a resolution of 1920 by 1200 pixels. To increase the accuracy of the experiments, we adjusted the height of the display and its distance from the participants within the recommended range of the eye tracker. Also, prior calibration was done for each participant. [Fig. 6\(a\)](#) illustrates the setup used for the study.

We showed slides from a pool of 28 distinct layouts, based on 15 data sources. To avoid participant eye fatigue and ensure the most accurate results, the number of tests per participant was limited to 7 and a short break was provided after collection of data for each slide. Participants were instructed to observe the slides, on the basis of whether the content was interesting to them, and consume the content (i.e., read or observe the text and images). Participants were free to look at the screen in any manner they preferred, picking any point on the screen to start and observing for any duration. The measurements were stopped when the user looked beyond the boundaries of the screen, which corresponds to attending for the exit tile in the display foraging model. We took measures to increase the reliability of the results. Firstly, we considered a tile perused only when the eye fixation within its boundaries was above a certain threshold [43], to account for situations – frequently observed during preliminary experiments – wherein a participant would glance across the whole display before starting to explore the content of individual tiles [42]. In addition, we encouraged participants to consider all content of a slide, thereby increasing the chances of foraging times being collected for every tile in each layout.

<sup>5</sup> All participating under informed consent.



**Fig. 6.** Setup of the user studies: (a) a 24" display, the Tobii Eye-tracker 4C and a laptop running a web browser as well as the display foraging system for the gaze-tracking-based study; (b) a 55" TV, a Microsoft Kinect v2 and a workstation to log the data for the field study.



**Fig. 7.** Correlation between the total foraging time predicted by the display foraging model and that observed through gaze tracking in the user study.

### 6.3. Results

We aim at evaluating the accuracy of our proposed system. To do this, we compare the total foraging time *predicted* by the optimizer through the display foraging model against the *observed* total foraging time obtained by measuring eye gaze. We then fit a linear model by means of linear regression to determine how well the two values agree with each other. Accordingly, Fig. 7 shows the correlation between the predicted and observed values for total foraging time. Individual data points in the figure refer to slides with a given layout, averaged over all related measurements in the user study.

We calculated the  $R^2$  statistical measure of the data, which indicates how well the model explains the variability of the observations within the range 0%–100%. The model achieved an  $R^2$  value of 66%, which denotes a strong correlation between the values. This figure clearly shows the model's particularly good fit with the experimental data for observed foraging times between 15 and 25 s, corresponding to a predicted foraging time between 35 and 60 s. These values correspond to layouts composed of many tiles (4–7), with more than two of them containing long text passages. Also, the results demonstrate that the foraging-based optimization does accommodate variance in the user behavior, especially since the gain parameters were not obtained from the same sample used in our experiments.

In light of the complexity of modeling human behavior, our model proved reasonably good at predicting the time for reading multiple content elements. This is challenging in the scenario considered, in that a user can decide to stop paying attention to the pervasive display at any time before reading all the tiles in the slide. Despite that, the prediction of foraging time is accurate, implying that the IFT-based model effectively characterizes user behavior to optimize content.

### 7. Field study

We also conducted a field study to assess the ecological validity of our approach in real settings [25,56]. In doing so, we followed the guidelines by Alt et al. [55] to determine how interested people are in slides generated by the proof-of-concept system. Specifically, we compared our system to a baseline, a typical digital signage application showing a slideshow wherein each slide corresponds to a single piece of content, similar to Veenstra et al. [13].

The baseline created a playlist of content items selected without any criteria, whereas our system generated a sequence of best-possible slides, each showing a tiled layout with multiple content items.

In both conditions, we used a 55" LCD display on a floor stand at the eye level of the target audience to ensure that content would be readable to passers-by. The content was updated daily and covered diverse topics, such as news, lectures,

**Table 1**  
User behavior in the field study for the two systems.

Solution	Samples	Ignores	Glances	Reads & Stays
Baseline	885	95.04%	4.17%	0.79%
IFT-based	871	90.56%	7.52%	1.92%

weather, and posts from Twitter. We attached a Microsoft Kinect v2 to the top of the display to capture depth-frame images of people passing in front of the display<sup>6</sup> without the possibility of identifying them. The study was deployed in the lobby of a building at our university, a meeting point or hub through which people pass for diverse purposes: to go to lunch, wait for friends or before classes, and so on. The lobby had four further displays at the time of the field study, three non-interactive displays attached to the top of a wall and one interactive display placed on a pillar. Fig. 6(b) shows the setup employed in the field study.

We collected data during workdays from 9:00am to 3:00pm for four weeks. We deployed our proposed IFT-based system and the baseline in the same exact location in the lobby. The baseline was deployed first, followed by our system. There was a 4-day pause between the deployments to prevent passers-by getting used to the display. We recorded 120 h of video and manually analyzed them so as to observe the behavior of passers-by, according to the methodology of Müller et al. [3,57].

We are aware that our study has limitations: the installation of the display could have caused the novelty effect during the first days, and technical limitations prevented the installation of the display in a place more noticeable. These limitations could be overcome by running the field study for a longer period of time and also by performing in-situ observations to corroborate the data obtained from the Kinect. In addition, both solutions could be deployed at the same time to allow better assessment. However, despite such limitations, the obtained results are significant and meaningful for comparing the two considered solutions.

### 7.1. Findings

Table 1 shows the results for the two systems considered: the baseline and our IFT-based system. These results correspond only to the passers-by caught in the field of view of the Kinect. In our analysis, we only consider those passers-by detected at a distance of no more than 2 m.

From the videos, we analyzed three patterns of user behavior: (1) passers-by walking without looking at the display, (2) passers-by glancing the screen while walking, and (3) passers-by standing in front of the screen and reading. We regard as an indicator of interest when users stop in front of the screen for more than 3 s, while users glance the content on a screen in less than 2 seconds [9].

The results show similarity between the two cases in the total number of people walking around in the area. As expected, there was a large percentage of passers-by walking without noticing the display, due to the known phenomenon of display blindness [5,56]. However, the results indicated a significant difference between the baseline and our system in terms of interest  $\chi^2(2, N = 1,756) = 13.1, p < 0.05, d = 1.98$ . In particular, the proposed system almost doubled the number of people glancing at the display, compared to the baseline  $\chi^2(1, N = 1,756) = 8.05, p < 0.05, d = 1.96$ .

We observed several situations while analyzing the videos when our system was running. The time users spent on looking at the screen was 5 seconds on the average, a similar value to the one reported in [9,25]. However, there were some cases where users attended to the display for more than 30 seconds. This did not occur for the baseline at all, which is also consistent with the results in Veenstra et al. [13]. Also, a user standing in front of the screen enticed people passing by to glance at the screen while walking, similar to what has been observed for interactive displays [12]. We noticed also that some passers-by walking in front of the screen came back when they noticed a change in the content. A few people took pictures of the screen before walking away.

## 8. Discussion and future work

IFT is based on rational analysis, the idea of studying human behavior by estimating the best achievable performance by assuming that users are fully rational. This notion has been implemented in bounded agents and computational rationality [43,58], wherein reinforcement learning is used to obtain behavioral policies for very general cases. In our case, the benefit of using IFT is that it incurs a lower computational overhead than do models for reinforcement learning.

Although our proof-of-concept optimizer does search a significant number of candidate solutions – as many as the computational budget allows – the optimal content assignment may not necessarily be found by the random multi-start method. However, our experience is that there is a large number of good-enough slides that can be quickly identified. Nevertheless, we are aware that our approach based on decomposition could be replaced with one applying joint generation and assignment of content. We leave this option as future research.

We are currently exploring options to further improve model accuracy in several respects. Firstly, our current model assumes that the parameters of the IFT functions are known for all content options. To improve on the results, we can

<sup>6</sup> Following applicable laws, a sign was posted in the lobby to notify people about the study, their rights, and the anonymity of collected data.

**Table A.1**

Average values obtained from the Likert-scale ratings in the online survey.

Category	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
Q1	3.81	3.06	3.11	3.66	3.16	3.70	3.75	2.95	3.23	3.32	3.29	3.38	3.34	3.28	3.23
Q2 (768 × 840)	3.24	3.48	2.17	2.58	2.34	2.55	2.41	2.09	2.26	2.60	2.30	3.06	3.53	3.42	3.14
Q2 (640 × 400)	3.31	3.28	2.12	2.60	2.20	2.53	2.47	2.00	2.32	3.56	2.17	2.99	3.64	3.56	3.25
Q2 (960 × 1200)	3.25	3.42	2.25	2.50	2.26	2.63	2.55	2.02	2.32	2.69	2.19	3.19	3.44	3.48	3.12
Q2 (1920 × 360)	3.34	3.34	2.18	2.75	2.29	2.79	2.58	2.06	2.41	2.94	2.29	3.26	3.50	3.13	2.47

infer such functions by employing a more sophisticated approach through, for instance, gaze-tracking data or the use of collaborative filtering. Secondly, we assumed that users start reading the display at a given tile (e.g., the one in the top-left corner) and that there is a single landing point for gaze. However, any starting point might be determined – e.g., on the basis of an empirical study [56]. In one alternative, a saliency model [59] could be used to compute a distribution for the most likely starting points given the actual content of the display. Finally, we can extend our approach to improve human perception. For instance, we could support users' finding of particular content in specific locations on the display. To do this, we could employ a temporal smoothness factor for ensuring that the position of an item does not change much from one update to another. With another option, we could use a cognitive model of visual learning [60] to estimate the costs of changing the position of a certain item.

## 9. Conclusion

Our work addressed the challenging problem of engaging users with the content shown by a pervasive display. A key decision was to focus on displays showing multiple content options at a time by arranging them into tiled layouts. In particular, we focused on non-interactive displays since it is more challenging to entice user's attention. In this context, we introduced a model that applies information foraging theory to design interesting content for pervasive displays. Such a novel approach leverages a behavioral characterization of a rational but busy user to describe the information gain associated to a display. Following that, our goal was to select the most interesting content and arrange it into a tiled layout. We devised a proof-of-concept system, evaluated its computational performance, and carried out an empirical study. A controlled user study through gaze tracking demonstrated that the display foraging model accurately characterizes the time a user spends on looking at a tiled display. Moreover, a field study showed that the proposed IFT-based system outperformed a typical solution for digital signage in terms of the number of users looking at the display. Specifically, the number of glances with our system nearly doubled compared to the baseline.

Indeed, the proposed approach works without user intervention or external device, hence it eases the deployment of this solution. Moreover, these findings can be considered a starting point to further explore information foraging theory in this context.

## Acknowledgments

This work was partially supported by the Academy of Finland (grant number 299222) and the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant number 637991). We would like to thank the CSC-IT Center for Science and the Aalto Science-IT project for provisioning the computational resources used for the evaluation.

## Appendix. Results of the survey-based study

In the study presented in Section 6.1, we asked participants to rate content items based on a 5-point Likert scale. We generated 60 tiles by using 4 distinct tile sizes to display 15 different content items. Each tile corresponds to a set of questions in the survey, including: one that characterizes the interest level (“I found the shown content interesting”, referred to as Q1); another that relates to the viewing time (“I would spend over 15 s to examine this content”, referred to as Q2). Contents are grouped into different categories: menus from cafeterias (C1, C4), dissertations (C2), tweets from university accounts (C3, C5, C8), weather (C6, C7), national news (C9, C10), events (C11), university news (C12, C15), and faculty news (C13, C14). We processed the responses and computed the average values reported in Table A.1. Note that responses to Q1 were averaged by category irrespective of the tile size, while those to Q2 were averaged by categories for tiles with the same size (shown within parenthesis in the table).



## References

- [1] S. Clinch, N. Davies, A. Friday, C. Efstratiou, Reflections on the long-term use of an experimental digital signage system, in: *Proceedings of the 13th International Conference on Ubiquitous Computing, UbiComp '11*, 2011, pp. 133–142.
- [2] T. Ojala, et al., Multipurpose interactive public displays in the wild: Three years later, *Computer* 45 (5) (2012) 42–49.
- [3] J. Müller, J. Exeler, M. Buzeck, A. Krüger, Reflectivesigns: Digital signs that adapt to audience attention, in: *Pervasive Computing*, 2009, pp. 17–24.
- [4] N. Davies, S. Clinch, F. Alt, Pervasive displays: Understanding the future of digital signage, *Synth. Lect. Mob. Pervasive Comput.* 8 (1) (2014) 1–128.
- [5] J. Müller, et al., Display blindness: The effect of expectations on attention towards digital signage, in: *Pervasive Computing*, 2009, pp. 1–8.
- [6] J. Patterson, S. Clinch, SlideTalk: Encouraging user engagement with slideshow displays, in: *Proceedings of the 7th ACM International Symposium on Pervasive Displays, PerDis '18*, 2018, pp. 4:1–4:7.
- [7] T. Lindén, T. Heikkinen, V. Kostakos, D. Ferreira, T. Ojala, Towards multi-application public interactive displays, in: *Proceedings of the 2012 International Symposium on Pervasive Displays, PerDis '12*, 2012, pp. 9:1–9:5.
- [8] S. Clinch, J. Alexander, S. Gehring, A survey of pervasive displays for information presentation, *IEEE Pervasive Comput.* 15 (3) (2016) 14–22.
- [9] E.M. Huang, A. Koster, J. Borchers, Overcoming assumptions and uncovering practices: When does the public really look at public displays? in: *International Conference on Pervasive Computing*, 2008, pp. 228–243.
- [10] Y. Taniguchi, Content scheduling and adaptation for networked and context-aware digital signage: A literature survey, *ITE Trans. Media Technol. Appl.* 6 (1) (2018) 18–29.
- [11] N. Davies, et al., Personalisation and privacy in future pervasive display networks, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14*, 2014, 2357–2366.
- [12] P. Peltonen, et al., It's Mine, Don't Touch!: interactions at a large multi-touch display in a city centre, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2008, pp. 1285–1294.
- [13] M. Veenstra, et al., Should public displays be interactive? Evaluating the impact of interactivity on audience engagement, in: *Proceedings of the 4th International Symposium on Pervasive Displays, PerDis '15*, 2015, 15–21.
- [14] S. Izadi, H. Brignull, T. Rodden, Y. Rogers, M. Underwood, Dynamo: A public interactive surface supporting the cooperative sharing and exchange of media, in: *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology, UIST '03*, 2003, pp. 159–168.
- [15] T. Linden, T. Heikkinen, T. Ojala, H. Kukka, M. Jurmu, Web-based framework for spatiotemporal screen real estate management of interactive public displays, in: *Proceedings of the 19th International Conference on World Wide Web, WWW '10*, 2010, pp. 1277–1280.
- [16] J. Müller, A. Krüger, How much to bid in digital signage advertising auctions? 2007.
- [17] J. Müller, A. Schlottmann, A. Krüger, Self-optimizing digital signage advertising, 2007.
- [18] J. Müller, A. Krüger, User profiling for generating bids in digital signage advertising auctions, *UBIDEUM 2007* (2007) 46.
- [19] M. Sethi, M. Lijding, M. Di Francesco, T. Aura, Flexible management of cloud-connected digital signage, in: *Proceedings of the 12th International Conference on Ubiquitous Intelligence and Computing, UIC 2015*, 2015.
- [20] V. Cheung, S. Scott, Proxemics-based visual concepts to attract and engage public display users: Adaptive content motion and adaptive user shadow, in: *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces, ISS '16*, 2016, pp. 473–476.
- [21] E. Lösch, F. Alt, M. Koch, Mirror, Mirror on the Wall: Attracting Passers-by to Public Touch Displays With User Representations, in: *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces, ISS '17*, 2017, pp. 22–31.
- [22] H. Brignull, Y. Rogers, Enticing people to interact with large public displays in public spaces, *Proceedings of INTERACT*, vol. 3, 2003, pp. 17–24.
- [23] D. Michelis, J. Müller, The audience funnel: Observations of gesture based interaction with multiple large displays in a city center, *Int. J. Hum. Comput. Interact.* 27 (6) (2011) 562–579.
- [24] A. Oulasvirta, User interface design with combinatorial optimization, *Computer* 50 (1) (2017) 40–47.
- [25] C. Parker, M. Tomitsch, J. Kay, Does the public still look at public displays? A field observation of public displays in the wild, *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2 (2) (2018) 73:1–73:24.
- [26] P. Pirolli, Information foraging theory: Adaptive interaction with information, 2007.
- [27] P. Pirolli, S. Card, Information foraging in information access environments, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1995, pp. 51–58.
- [28] J. Lawrance, R. Bellamy, M. Burnett, Scents in programs: Does information foraging theory apply to program maintenance? in: *Visual Languages and Human-Centric Computing*, 2007. VL/HCC 2007. IEEE Symposium on, 2007, pp. 15–22.
- [29] N. Niu, A. Mahmoud, G. Bradshaw, Information foraging as a foundation for code navigation (NIER track), in: *Proceedings of the 33rd International Conference on Software Engineering*, 2011, pp. 816–819.
- [30] P. Pirolli, W.-T. Fu, SNIF-FACT: A model of information foraging on the world wide web, *User Modeling* 2003 (2003) 146–146.
- [31] K. Verbert, E. Duval, J. Klerks, S. Govaerts, J.L. Santos, Learning analytics dashboard applications, *Am. Behav. Sci.* 57 (10) (2013) 1500–1509.
- [32] P. Pirolli, S.K. Card, M.M. Van Der Wege, Visual information foraging in a focus+ context visualization, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2001, pp. 506–513.
- [33] H. Liu, P. Mulholland, D. Song, V. Uren, S. Rüger, Applying information foraging theory to understand user interaction with content-based image retrieval, in: *Proceedings of the Third Symposium on Information Interaction in Context*, 2010, pp. 135–144.
- [34] L. Azzopardi, Modelling interaction with economic models of search, in: *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, 2014, pp. 3–12.
- [35] J.C. Goodwin, T. Cohen, T. Rindfleisch, Discovery by scent: Discovery browsing system based on the information foraging theory, in: *2012 IEEE International Conference on Bioinformatics and Biomedicine Workshops*, 2012, pp. 232–239.
- [36] N. Lesh, J. Marks, A. McMahon, M. Mitzenmacher, Exhaustive approaches to 2d rectangular perfect packings, *Information Processing Letters* 90 (1) (2004) 7–14.
- [37] A. Borning, R.K.-H. Lin, K. Marriott, Constraint-based document layout for the web, *Multimedia Systems* 8 (3) (2000) 177–189.
- [38] E. Ch'ng, D.C.L. Ngo, Screen design: a dynamic symmetry grid based approach, *Displays* 24 (3) (2003) 125–135.
- [39] P. Pirolli, S. Card, Information foraging in information access environments, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1995, pp. 51–58.
- [40] P. Pirolli, S. Card, Information foraging, *Psychological Review* 106 (4) (1999) 643.
- [41] E.L. Charnov, Optimal foraging, the marginal value theorem, *Theor. Popul. Biol.* 9 (2) (1976) 129–136.
- [42] R.W. Baloh, A.W. Sills, W.E. Kumley, V. Honrubia, Quantitative measurement of saccade amplitude, duration, and velocity, *Neurology* 25 (11) (1975) 1065–1065.
- [43] X. Chen, G. Bailly, D.P. Brumby, A. Oulasvirta, A. Howes, The Emergence of Interactive Behavior: A model of rational menu search, in: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI '15*, 2015, pp. 4217–4226.
- [44] S. Boyd, S.-J. Kim, L. Vandenbergh, A. Hassibi, A tutorial on geometric programming, *Optimization and Engineering* 8 (1) (2007) 67–127.



- [45] M.C. Riff, X. Bonnaire, B. Neveu, A revision of recent approaches for two-dimensional strip-packing problems, *Eng. Appl. Artif. Intell.* 22 (4) (2009) 823–827.
- [46] A. Kaul, S. Aggarwal, A. Gupta, N. Dayama, M. Krishnamoorthy, P.C. Jha, Optimal advertising on a two-dimensional web banner, *Int. J. Syst. Assur. Eng. Manage.* (2017).
- [47] K. Todi, D. Weir, A. Oulasvirta, Sketchplore: Sketch and explore with a layout optimiser, in: *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, 2016, pp. 543–555.
- [48] M. Sethi, E. Oat, M. Di Francesco, T. Aura, Secure bootstrapping of cloud-managed ubiquitous displays, in: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '14*, 2014, pp. 739–750.
- [49] S. Clinch, M. Mikusz, M. Greis, N. Davies, A. Friday, Mercury: An application store for open display networks, in: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '14*, 2014, pp. 511–522.
- [50] Bootstrap toolkit, Get bootstrap, December 2018, <http://getbootstrap.com/>.
- [51] Node.js foundation, Node.js javascript runtime, October 2017, <https://nodejs.org>.
- [52] C. McKinstry, Socket.io: Realtime application framework, December 2018, <https://github.com/socketio/socket.io>.
- [53] R. Martí, J.A. Lozano, A. Mendiburu, L. Hernando, Multi-start methods, in: *Handbook of Heuristics*, 2016, pp. 1–21.
- [54] S. Consolvo, M. Walker, Using the experience sampling method to evaluate ubicomp applications, *IEEE Pervasive Comput.* 2 (2) (2003) 24–31.
- [55] F. Alt, S. Schneegaß, A. Schmidt, J. Müller, N. Memarovic, How to evaluate public displays, in: *Proceedings of the 2012 International Symposium on Pervasive Displays, PerDis '12*, 2012, pp. 17:1–17:6.
- [56] N.S. Dalton, E. Collins, P. Marshall, Display blindness? Looking again at the visibility of situated displays using eye-tracking, in: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 2015, pp. 3889–3898.
- [57] J. Müller, R. Walter, G. Bailly, M. Nischt, F. Alt, Looking glass: a field study on noticing interactivity of a shop window, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012, pp. 297–306.
- [58] S.J. Gershman, E.J. Horvitz, J.B. Tenenbaum, Computational rationality: A converging paradigm for intelligence in brains, minds, and machines, *Science* 349 (6245) (2015) 273–278.
- [59] L. Itti, C. Koch, E. Niebur, A model of saliency-based visual attention for rapid scene analysis, *IEEE Transaction on Pattern Analysis and Machine Intelligence* 20 (11) (1998) 1254–1259.
- [60] J.P. Jokinen, S. Sarcar, A. Oulasvirta, C. Silpasuwanchai, Z. Wang, X. Ren, Modelling learning of new keyboard layouts, in: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2017, pp. 4203–4215.