Artificial Intelligence for Engineering Design, Analysis and Manufacturing

cambridge.org/aie

Research Article

Cite this article: Zhang C, López-Parra M, Chen J, Tian L (2019). CoStorm: a term map system to aid in a collaborative ideation process. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **33**, 247–258. https://doi.org/10.1017/S0890060418000215

Received: 10 December 2016
Revised: 22 August 2018
Accepted: 26 August 2018
First published online: 15 October 2018

Key words:
Computer-aided conceptual design; decision-making; electronic brainstorming; ideation

Author for correspondence:
Ling Tian,
E-mail: tianling@mail.tsinghua.edu.cn

CoStorm: a term map system to aid in a collaborative ideation process

Chengwei Zhang[1], Marcelo López-Parra[2], Junyu Chen[1] and Ling Tian[1]

[1]Beijing Key Laboratory of Precision/Ultra-precision Manufacturing Equipments and Control, Department of Mechanical Engineering, Tsinghua University, Beijing, China and [2]Unidad de Alta Tecnología, Facultad Ingeniería, UNAM, Queretaro, Mexico

Abstract

The decisions made during the early stages of a design process have a huge impact on a product. Owing to the explosion of preliminary ideas, however, designers easily lose track of important ideas and significant information and end up being buried in a pile of plain words. Failing to locate an idea in the context of idea generation makes it difficult to generate new ideas or take optimized decisions. In this study, the authors propose the term map approach to provide a complete bird's eye view of all ideas, which is a higher-dimension graphical representation that helps in inspiring ideas and making decisions among design team members. A software application named CoStorm is developed. Through the case study of the cash-flattener module, which is a crucial component of an automated teller machine, this method is found to contribute in facilitating the ideation and decision-making progress.

Introduction

Product design is an iterative and complex decision-making process (Hsu and Woon, 1998). The decisions made during the early stages of a design process have a huge impact on the product. It is estimated that 80%–90% of the life-cycle design costs (including fabrication, construction, energy, maintenance, and disposal) are determined in the first 10%–20% of the design phase (Hoover and Jones, 1991). In the early stage of design, there are relatively few constraints. The design team focuses on researching user's needs and generating creative conceptual solutions. Brainstorming is widely used for idea generation during these preliminary stages. It encourages wild and imaginative ideas; the more, the better. It also builds a sound basis for the embodiment, detail design, and evaluation stages that follow (MacCrimmon and Wagner, 1994). The explosion of preliminary ideas entails more possibilities of innovative products, along with challenges when making decisions among varied alternatives.

Usually, there are several concerns that design teams encounter when they creatively analyze needs, generate ideas, produce conceptual solutions, and make decisions. Normally, numerous ideas are brought about during the ideation phase, but many times, designers lose track of what is important. Significant information ends up being buried in a pile of plain words. On the other hand, it is also common to find various design groups working on the same project at different geographical locations, making it more difficult to keep everyone involved on the same page. The design process also implicates constant ideation, testing, and evaluation activities. Therefore, it becomes difficult to obtain a complete bird's eye view of all the ideas generated in a particular session or a period. Failing to locate an idea in the context of all ideas makes it difficult to develop design solutions and take optimized decisions.

The record of the ideation process is linear, when projected on a timeline. In this study, we propose an approach to turn it into a higher-dimension graphical representation that helps in improving the ideation and decision-making process of design teams. Based on the method, a software application named CoStorm, developed in this study, is described and then used in designing a cash-flattener module, which is part of an automated teller machine (ATM). Thus, the effectiveness of the proposed approach is demonstrated.

Related work

Stimuli for ideation

Stimuli are defined as all information that is provided or used for inspiring new ideas (Howard *et al*., 2011), and it is commonly agreed that the presence of stimuli are effective for inspiring more ideas (MacCrimmon and Wagner, 1994; Kletke *et al*., 2001). Howard *et al*. (2011) categorized stimuli by source and retrieval. A lot of research has been carried out on modeling or using external stimuli, such as design by analogy using biomimetic and artificial sources (Chakrabarti *et al*., 2005; Nagel *et al*., 2010; Murphy *et al*., 2014; Pal *et al*., 2014). Internally sourced stimuli, on the other hand, proved to be effective in terms of rate of idea generation and quantity of ideas (Howard *et al.*, 2011). Jin and Benami (2010) argued that design entities, as the result of creation, are also the catalysts for further creation. Srinivasan *et al.* (2015) studied the use of internal analogy as stimulus in conceptual design. These authors concluded that stimulus proved to be a good trigger to inspire creative ideas.

The use of clues or hints in the design environment has been emphasized as a rich source of new ideas and it has also been recognized as an important method to overcome design fixation (Cardoso and Badke-Schaub, 2011). AIDA (analysis of interconnected decision areas) matrix offers a strategic graph method, contributing in organizing decision space and in identifying the dependency among decision areas (Weas and Campbell, 2004). However, due to a lack of prior knowledge, it is difficult to provide a structure to a problem and target domains when using AIDA. To provide a guided internal stimulus, Linsey *et al.* (2008) proposed the WordTree method to provide guidance in a particular field of design-by-analogy. Creating WordTrees to represent the problem helps in distinguishing the analogies and analogous domains. When using created ideas or concepts as stimuli or alternative solutions, working-memory capacity restrains the information a designer can maintain and leaves fixated features in design unnoticed (Youmans, 2011; Youmans and Arciszewski, 2014). Design fixation goes against inspiration and makes concept generation and decision making increasingly difficult.

Although lots of research has been done in representing and using different types of stimuli for inspiration, it is still a demanding task to try to organize internal stimuli with flexibility. The authors believe that inspiring designers with the ingenious use of their own ideas is a topic that has not been fully explored.

Electronic brainstorming system

The brainstorming technique is widely used for generating ideas and original solutions within design teams. Osborn (1953) defines brainstorming as follows: "To practice a conference technique by which a group attempts to find a solution for a specific problem by amassing all the ideas spontaneously contributed by its members." Gallupe and Cooper (1993) referred that compared to conventional oral brainstorming, electronic brainstorming enabled dispersed groups to interact more closely. It dealt with the shortcomings of "evaluation apprehension", "free riding", and "production blocking" issues that are common in conventional brainstorming (Diehl and Stroebe, 1987; Sutton and Hargadon, 1996; Elias *et al.*, 2011). These latter authors reported that electronic brainstorming improved the productivity of larger design groups, but even in the context of electronic brainstorming, clustering the ideas or solutions was found to be a very time-consuming task. Furthermore, it required a highly experienced organizer, who could put an idea or solution in the proper position on a map. Chandrasegaran *et al.* (2013) reviewed both product design processes based on knowledge and support tools that currently exist to perform brainstorming. They referred that, with huge numbers of raw data available to designers, the challenge for decision making is to "sift through this data and make sense of it."

Dennis *et al.* (2013) brought the priming phenomenon, which alters subsequent behavior by a subconscious concept in working memory, into group behavior and idea creation. Experiments were conducted using a computer game designed to improve creativity through priming. Researchers referred that priming helped groups in generating more ideas that were more imaginative. However, they did not discuss any methods that could help to construct priming for ideation. To visualize the structure of design rationale, Bracewell *et al.* (2009) proposed a design rationale editor, and further used it to create a function analysis diagram (Aurisicchio *et al.*, 2013). For the "mashup" (Vallecillos *et al.*, 2014) problem when brainstorming among multiple users, Moulin *et al.* (2016) proposed three dimensions (resource, video, and data persistence) concerned with the capitalization of brainstorming activities in a remote context. They also implemented a suite of applications. Multi-dimensional data during brainstorming are collected and displayed on an interactive whiteboard. However, data gathered were not well organized and reported. Post-it stickers showed the concept, without capabilities for supplying the details when required.

Yuizono *et al.* (2014) applied gamification to electronic brainstorming. They implemented an electronic brainstorming system based on a blog software, with a presumption of brainstorming in the form of text. Their experiment showed how three types of fun as gamification features could improve idea generation. Chiu and Tomimatsu (2013) performed research on effective strategy dealing with "casual data" in social network service for the fuzzy front end. They reported that combining "different information that appeared separate at the outset" could facilitate the divergent thinking progress. They also proposed a system where producers and users can publish questions and suggest opinions respectively and recursively. However, the hierarchical blog and its comments have not been summarized yet. At the same time, the potential relations between blogs and comments are still hidden.

Published evidence reviewed by the authors of this work suggests that the electronic brainstorming method has proved to be effective. Research has been carried out for improving the process of idea creation. However, there is still a lot of work to do and questions to be answered to completely take advantage of the flexibility of the digital design space. For example, how could an overview of the brainstorming results help inspire more ideas and contribute to the decision-making process?

Commercial software

A software package that can facilitate the realization of all activities that are carried out all along the development of a design project does not exist. However, there are several packages that are useful to support different stages and aspects of the design process.

Freemind[1] is one of the most popular mind-mapping tools. Designers enter conceptual ideas to this computer tool and are then guided by the software to devise a mind map. It supports individual designers. In the context of collaborative design, this type of software meets the challenges on how to share one's ideas with other designers, minimizing potential misunderstandings among team members, and preventing ideas from being obstructed by any individual.

Slack[2] and HipChat[3] are real-time messaging platforms; they make use of instant messages to create a smoother design process. They collect basic data, such as texts, and capture multi-media, such as voices, images, and videos. All of them are searchable; however, all data are displayed linearly. Lacking an overview of the brainstorming process makes it difficult to take decisions for all the alternatives.

[1]http://freemind.sourceforge.net/wiki/index.php/Main_Page
[2]https://slack.com
[3]https://www.hipchat.com

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

Stormboard[4] is an online brainstorming and collaboration tool. It replaces the whiteboard with real-time online sticky notes. Designers can "quickly add, organize, discuss, vote, and act on the best ideas." However, ideas categorized in level one are still in a flat namespace. The relations between ideas are not shown clearly. The latter is also true for the case of Mural.[5]

Commercial software packages provide support to some design activities but there is still a lot of work to do to assist the designer all along the design process. Designers do not enter their ideas directly. They transform descriptive sentences of ideas into short words or phrases and then log them into the map. Designers usually have to explain the words to other designers, repeatedly if required; this fact seriously interferes with the communication that exists between the idea generator and the designer receiving the new concept. Furthermore, lacking a bird's eye view of ideas makes it difficult to locate one idea. This is not beneficial for inspiring new ideas and makes decisions from the existing design alternatives.

Aim and significance

In this study, we propose a term map method to facilitate the design process. Issues that we are seeking to deal with include the following: how to collect ideas from geographically dispersed designers? How to generate a term map? How would a term map help in inspiring new ideas? How would a term map contribute to the decision-making progress? By using the software application CoStorm, developed during this research work, we were able to verify the effectiveness of the method with a practical case study concerning the design of a cash-flattener module, which is part of an ATM. The findings presented here could shed light on the idea generation and decision-making processes in large design groups such as in crowd-sourcing design projects.

Method and approach

A term is the keyword of an idea message, and a term map is a map of terms, showing the relations among terms. The term map reveals how designers develop their design concept divergently or progressively. The map is not only a stimulus for inspiring more ideas, but is also an overview of the brainstorming session. The term map method follows the logic flow shown in Figure 1. It starts with collecting distributed designers' idea messages. We model the idea message as a design state, thereby leveraging the computational assistance. After extracting the terms from an idea, three strategies are used to annotate the relation between a term and other terms, which are already in the map. With terms and relations added to the map repeatedly and recursively, the term map is constructed and developed. The term map shows how different ideas evolve, branch, and merge. Therefore, designers generate new ideas based on the previous ones. The three views of cluster of terms provide a clue to review the brainstorm session, and therefore, provide designers with related idea information when making a final decision. We developed a prototype system, named CoStorm, and applied it in a case study to verify how the method could help designers in making decisions.

[4]https://stormboard.com
[5]https://mural.com

Message modeling

In the early stage of the design process, design problems are mostly ill defined. There are just general and abstract descriptions instead of concrete requirements. Thus, in the beginning, designers make those requirements more specific and put forward the clusters of preliminary thoughts, mainly in the form of text. These ideas are generally quite diverging and under-structured; in order to have them processed by a computer, we propose new models to help in guiding the brainstorming and idea generation processes.

Within the brainstorming process, a design space is a set covering all information of the entire design process, including a time stamp and linguistic description. All points in this space represent all possible states explaining all details of the design process of a product's life cycle.

An idea, with all the information about it, defines a design state. It is a point in the design space and can be represented by a vector containing multi-dimensional information. For a better understanding of the relation between the design states, the model of a design state is enhanced with markers. First, a list of keywords, that is, terms, are figured out from the idea message. These terms compose the main features of a message. Then, the relation among terms from different design states is built.

Therefore, the design state is further defined as

$$\begin{aligned}\text{design\_state} = < \text{id, timestamp, speaker\_id} \\ \text{message, term\_list, relation\_list} >\end{aligned} \tag{1}$$

where term_list is a set of terms in the message, and relation_list is a set of relations between terms in the message and terms in other design states. Relation is defined as a triple tuple:

$$\text{relation} = < \text{sub, pred, obj} > , \tag{2}$$

where sub is a term in the term_list of the current design state, pred is a relation type, and obj is a term in the existing design states.

The model built for the design state for early conceptual design should be elaborated into a protocol for storage and communication. A concrete message content could be instantiated. Inspired by the graph concept in the graph theory of computer science, we find these mathematical structures as useful tools to model pairwise relations between objects (West, 2001) and a great abstract of such a complex situation. A graph is made up of vertices, which are connected by edges; each message is a vertex and the edge between vertices shows their connections. Thus, the information generated during the design process can be modeled as a graph, becoming the base for storage, calculation, and processing of the discussion that takes place during the idea generation sessions.

Term map generation

Design states are submitted by distributed or local designers. These states, following the model proposed, lie in the database without any relation revealed. Design states share a set of terms, which can be used to build a graph G_{tm} = (term, relation). The vertices, that is, terms, are now defined and ready to be used. To structure this process, the authors propose three different strategies to add every relation and then proceed to construct the map.

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

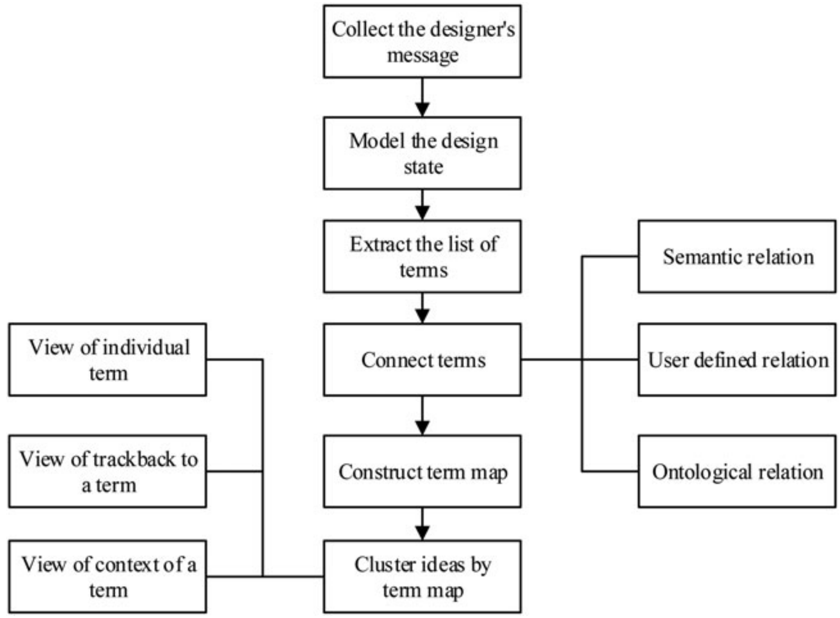

Fig. 1. Logic flow of the term map method.

Semantic relation

The message itself is the most direct source for collecting the relations. Basically, when terms appear in one sentence, they are likely to be correlated.

We define term sets, namely map_term_set = {term|term ∈ term_map} for the existing terms in the map and state_term = {term|term ∈ message} for terms in one design state. For a design state with more than one term, there are two types of relations between them: listing terms of juxtaposition and explaining terms in a logical sequence. The former is relatively weaker because it ignores the order, whereas the latter emphasizes the sequence.

Regarding states, we can have the following cases:

- A new state is defined when state_term ∩ map_term_set = ∅. As a design state describes a specific state, it should not cover a progressive process. Terms in states_term are regarded to be listed with weak connections. Therefore, terms would be pushed into the map_term_set separately, without connections added.
- When state_term ∩ map_term_set ≠ ∅ and map_term_set\states_term ≠ ∅, the design state is a progressive state based on the former. The terms $\{c_i|c_i \in$ states_term ∩ map_term_set$\}$ describe the precondition, whereas the terms $\{c_i'|c_i' \in$ map_term_set\states_term$\}$ talk about this design state. Thus, when pushing those c_i' to the term map, each of them relates to every element in $\{c_i\}$ separately.

Extra operations of designers are not required for this type of relations. Relations are extracted among terms from the messages themselves, without special intervention from designers. Every term in one design state would be pushed into the set by different rules according to its relation to the term map. In the method proposed, a design state containing no more than five terms is recommended, avoiding an over complex connection.

User-defined relation

For the latter case above, putting forward a design state with its precondition might be inefficient, because users would have to retype those precondition terms. In the continuous case, where one design state exists immediately after another, designers know what term this design state should follow, and this type of relation should be added directly. Thus, for the first type of user-defined relation, before putting out a message, designers could select a determined term on the map as a parent node. It would be annotated as "parent." Then every term in the state would be connected with the selected one separately as children nodes. This makes sense when designers decompose general terms into specific topics.

In the term-map-making strategy mentioned above, relations are built when a design state is put forward. However, in a discussion, there would be a lot of design states created independently. They are connected with no one because designers do not even know their relation and location in the term when proposing them. Organizing design states after they have been entered should be supportive. For the term map with some term clusters and some isolated terms, terms could be connected to be integrated with a cluster or to be constructed as a new cluster. When selecting two terms to be connected in the term map, a "connect request" of this connection is built and sent. After the relation in the term map is constructed, it is broadcast to all users for rendering the map shown in the browser. This contributes to the early design stage where users can hardly find the relations among their ideas, and later thoughts have to be made clearer.

Take the case in Figure 2 for example. *Message1* contains *Term1*. When it is put into the design space, *Term1* would be added to the term map as a separate node. When *Message2*, containing *Term1* and *Term2*, is entered, *Term1* would be recognized as an existing term, and *Term2*, as a new one, would be added to the term map with a connection with *Term1*.

Ontology relation

Strategies for making a term map in one design team's discussion have already been described. There is another type of knowledge source for constructing relations between terms. That is abstractly called the knowledge of enterprise, which is of great help in

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

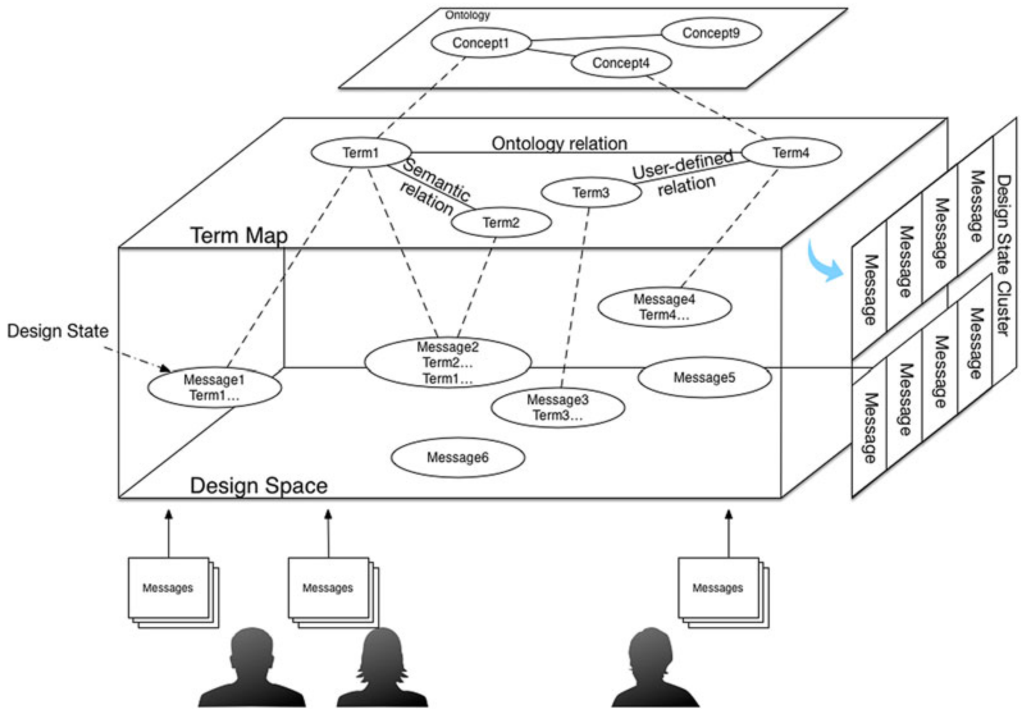

Fig. 2. Design space.

reusing design and product knowledge. However, knowledge, especially within a design group, is extremely vague to be used by individuals.

A method of storing and reusing the knowledge of enterprise is proposed based on an ontology. Ontology is defined as "explicit formal specifications of the terms in the domain and relations among them" (Gruber, 1993). It provides a common structure of information.

A general domain ontology O_{gd} is defined as a triple tuple for presenting explicit knowledge:

$$O_{gd} =< C, R, F > , \tag{3}$$

where C is the set of concepts, R is the set of relations between concepts, and F is the set of facts. Each statement is a triple tuple, which is $F\subseteq C\times R\times C$.

The domain ontology, shown in Figure 3, is regarded as an extended hierarchy tree. The "is-a" relation, that is, inheritance relation, constructs the hierarchy part of the ontology, in which one node has one parent node at most. When adding some relations with property relation in the ontology, nodes from two branches might be connected.

According to the theory on corpus statistics and lexical taxonomy (Jiang and Conrath, 1997), we calculate the information content (IC) of a concept as follows:

$$\mathrm{IC}(c) = \log^{-1}P(c), \tag{4}$$

where $P(c)$ is the possibility of a class to inherit directly or indirectly from it.

For a hierarchical structure, $P(c)$ is monotonic with the increase in the level of hierarchy. As all concepts extend from the root concept, $P(c_{\mathrm{root}}) = 1$ with $\mathrm{IC}(c_{\mathrm{root}}) = 0$.

Given the characteristic of monotonicity, the similarity between two concepts could be defined as follows:

$$\mathrm{hsim}(c_1, c_2) = \max_{c\in \mathrm{Sup}(c_1,c_2)}[\mathrm{IC}(c)] = \max_{c\in \mathrm{Sup}(c_1,c_2)}[-\log p(c)], \tag{5}$$

where $\mathrm{Sup}(c_1, c_2)$ is the set of concepts that derive both c_1 and c_2.

As the ontology is an extended tree, instead of a pure hierarchical tree, a connection might be added between two nodes on different branches with the relation of "relates_to." The similarity between concepts is defined as follows:

$$\begin{aligned}\mathrm{sim}(c_1, c_2) =& \mathrm{hsim}(c_1, c_2) + \alpha \max_{c\in \mathrm{Neighbor}(c_1)}[\mathrm{hsim}(c, c_2)]\\ &+ \beta \max_{c\in \mathrm{Neighbor}(c_2)}[\mathrm{hsim}(c, c_1)],\end{aligned} \tag{6}$$

where $\mathrm{Neighbor}(c_i)$ indicates the concept set whose element is connected to c_i with a non-inheritance relation.

Take the case in Figure 2, for example, *Concept1* and *Concept4* are in domain ontology. There happens to be two terms, *Term1* and *Term4*, mapped to the two concepts. As *Concept1* and *Concept4* are close to each other in ontology, a relation between *Term1* and *Term4* would be constructed.

Clustering by term map

Distance measuring for design states

As mentioned before, the design state is defined as a vector, $\vec{s} = (\mathrm{timestamp, termset}, \ldots)^T$. The design space is a set $\mathrm{ds} = \{\vec{s}\}$. To evaluate the relevance, the distance function is defined as $\mathrm{dist}(\vec{s_1}, \vec{s_2})$.

The time stamp t and the term set ts in a design state are counted. $f_t(\vec{s})$ gets the timestamp of design states, whereas

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

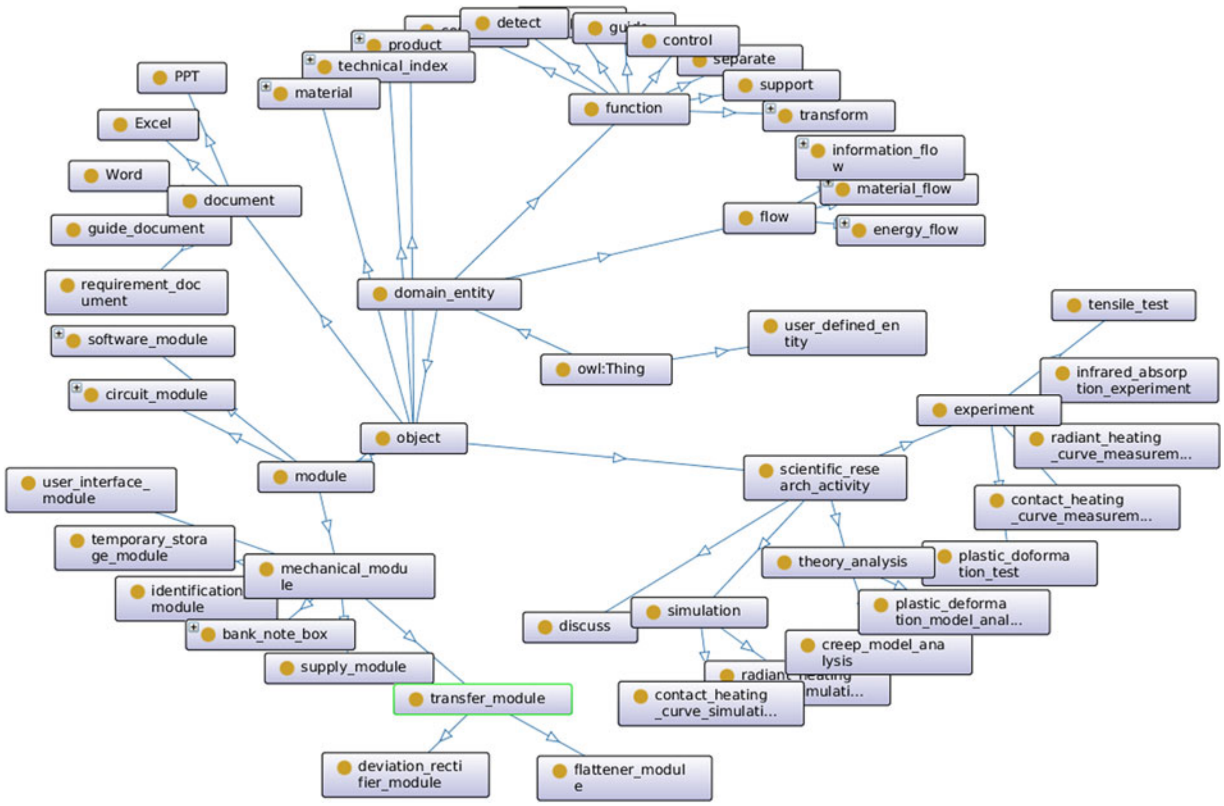

Fig. 3. Domain ontology of an ATM.

$f_{\text{ts}}(\vec{s})$ gets the term set of design state. $|f_{\text{ts}}|$ shows the number of elements in the term set.

f is defined as a map from a design state to the term set, as follows:

$$f(\vec{s}) = f_{\text{ts}}(\max_{s_i \in \text{Sup}(s)} f_t(\vec{s_1}) \cdot 1(|f_{\text{ts}}(\vec{s_1})| > 0)), \tag{7}$$

where Sup(s) is the design state that appear before the state s.

For two sets of terms ts_1, ts_2, the distance between them is defined as

$$\text{dist}(\text{ts}_1, \text{ts}_2) = \min_{t_i \in \text{ts}_1, t_j \in \text{ts}_2} \text{dist}(t_i, t_j), \tag{8}$$

where dist(t_i, t_j) is the distance between two terms in the term map.

Thus, the distance between design states s_1, s_2 could be defined as

$$\text{dist}(\vec{s_1}, \vec{s_2}) = \text{dist}(f(\vec{s_1}), f(\vec{s_2})). \tag{9}$$

For the operation of clustering for a certain term, the design states mentioning this term and those design states with a distance of zero with them should be figured out. Specifically, when selecting a determined term t, which is the core of the cluster, we search the design states mentioning t directly, forming the list $\text{ds}^t = [\text{ds}_1^t, \text{ds}_2^t, \ldots, \text{ds}_n^t]$. Each element acts as a core of the subcluster. Those design states belonging to the subcluster should be derived. For ds_i, we traverse design states by time and add every design state to the subcluster until a design state mentions another term. The algorithm is shown below.

Algorithm 1 Generation algorithm of term cluster

Input: select term t, set of Design States *dss*

Output: Design States Cluster based on Term Selected *dsc*

procedure GeneratingTermCluster(t, *dss*)	
cluster ← [], *subcluster* ← []	//Initialize
for design state *ds* in *dss* **do**	
if ds contains *any terms* **then**	//Find ds_i^t
if ds contains ^{t} **then**	//Core of subcluster
subcluster.insert(ds)	
else	
if *subcluster* $\neq \emptyset$ **then**	
cluster.insert(*subcluster*)	//A complete subcluster
subcluster = []	//Reset the subcluster
else	
if *subcluster* $\neq \emptyset$ **then**	//Core of subcluster exists
subcluster.insert(ds)	//Add a complete subcluster
return *cluster*	

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

View of design

The term map acts as a structured access to all design states. Three views are proposed to provide a better look at design state clusters in different levels:

(a) Individual. Every term in the term map is a representation of a collection of design states. Listing all the design states related to a certain term help the designers locate themselves and look into details of the brainstorm.
(b) Track. Designers might find one interesting term when learning where it comes from will contribute to the summary of the discussion and even inspire more ideas. Hence, the term in the map should be traced back to the origin. A path of this term is extracted, and all design states along the path would be organized. With a term t selected and the trace depth n set, a list of terms along the path $t_{path} = [t^1, t^2, \ldots, t^n]$ would be built. For each element in the list, the design state cluster belonging to it could be built. Therefore, designers could get a view of the term paths with their related design states.
(c) Neighbor. There is always one or some terms for core problems. The neighbor of this term gives out relatively complete information for the context. Thus, a neighboring field of the term should be pointed out, and those design states should be organized. At first, the neighboring field of a selected term should be constructed. When the radius r is set, a breadth-first traverse could be started. A set of terms $t_{neighbor} = \{t|dist(t, t_{center}) < r\}$ in the neighboring field could be constructed.

Prototype system

CoStorm is a web application developed in this work for collecting, discussing, and summarizing ideas. Based on the theory and method researched, the browser/server architecture prototype system was designed and developed using Python and HTML5. Tornado acts as the server and Bootstrap as the web framework. WebSocket in HTML5 is used as the protocol for the instant message communication. A NoSQL database, MongoDB, is applied for the storage of message log and term map. The architecture of CoStorm is shown in Figure 4.

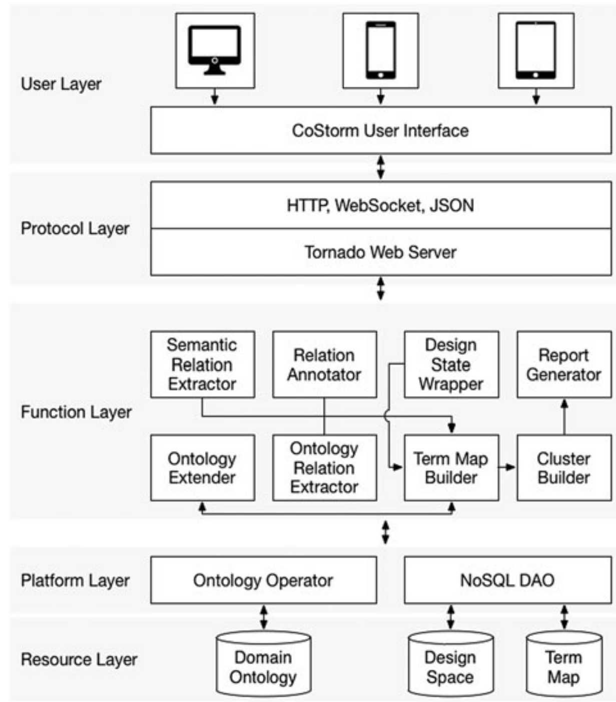

Fig. 4. Architecture of CoStorm.

Illustrative case

Design problem

ATMs are widely used around the world. It is reported[6] that in 2015, there were three million ATMs worldwide and the number of cash withdrawals reached above 8.6 million per year. ATMs contribute a lot in freeing labor in simple cash transactions.

There are three major components of an ATM that are used during a withdrawal transaction: dispenser cassette assembly, presenter unit, and dispenser pick module. The ATM, with multi-functions of withdrawal and deposition, is becoming more popular. To achieve continuous motion, banknotes are stored around a reel. Thus, the banknote in the cassette might be bent. This creates some problems such as getting stuck when the curly note is conveyed in the presenter unit. This leads to the design problem reported in this study, which is to minimize banknote jams caused by its flexural deformation.

[6]http://www.cutimes.com/2014/07/28/3-million-atms-worldwide-by-2015-atm-association?slreturn=1479833719

Design context

The university–industry co-development project was sponsored by GRG Banking Co., Ltd., one of the leading financial intelligent equipment manufacturers and solution providers. Throughout the development of the work, meetings were held on a regular basis to assure that needs and technical specifications were met by the design team.

The team comprised designers working in different geographical areas, over 2000 km away: Beijing in northern China and Guangzhou in southern China. Team members had different backgrounds and experiences. Engineers from the industry were good at routine solutions and specific product design processes, whereas researchers from the academe had numerous involvement in the analysis of general problems and the processes of producing innovative solutions. Geographical dispersal and varieties of background brought difficulties when brainstorming for the conceptual design stage.

Design steps

The design methodology SAPB (Pahl *et al.*, 2007) was followed during the realization of the project. As defined by the method, the steps of task clarification, conceptual design, embodiment design, and detail design were carried out. Alternative conceptual solutions were proposed and a final design was 3D modeled.

The modern ATM equipment is a highly modular system that operates under severe conditions. It has various major components that are responsible for correct functioning. A common problem encountered in ATMs is related to the use of the cash collector module (CCM). The CCM integrates six basic

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

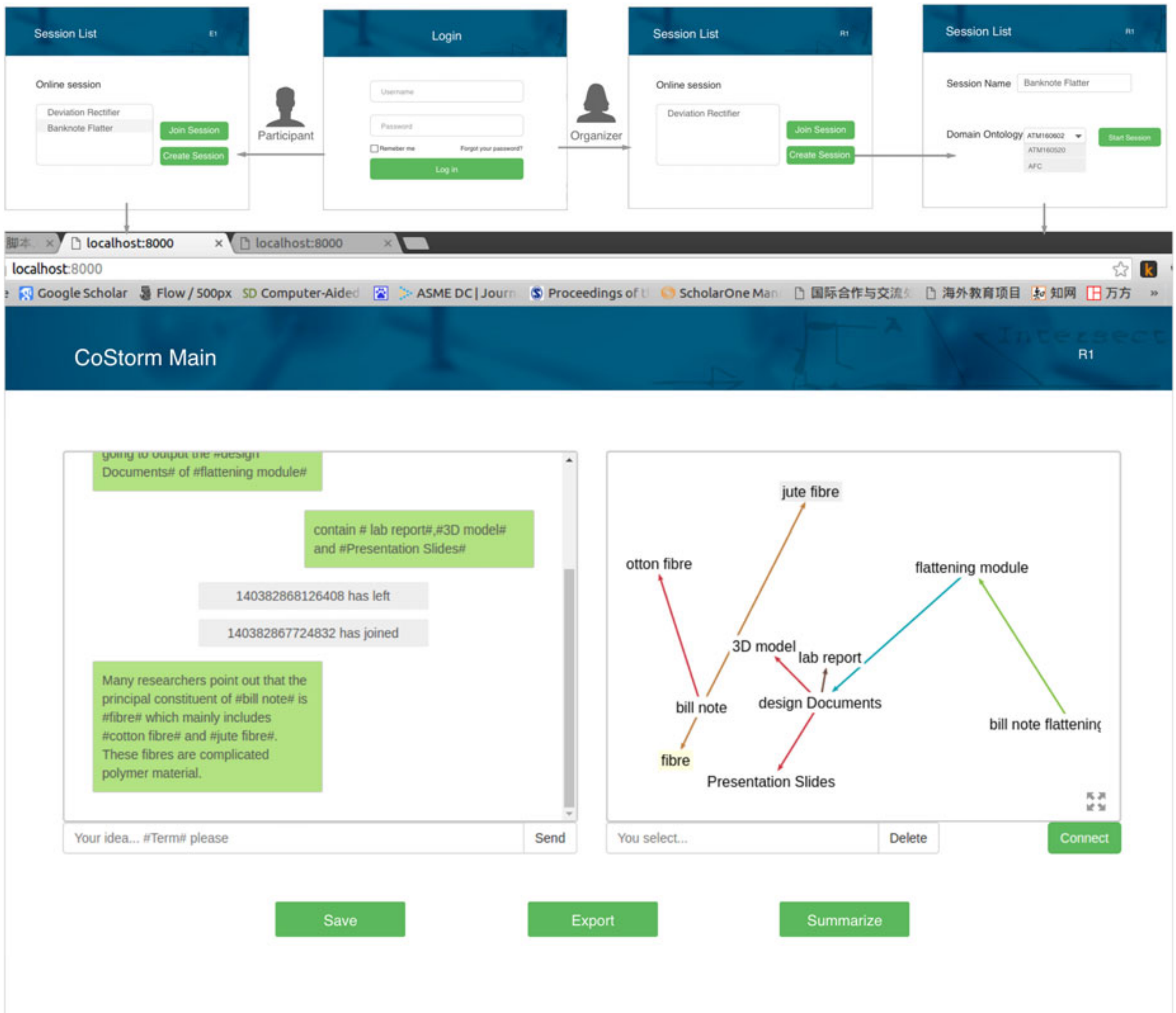

Fig. 5. Session operation.

operations: banknote insertion, conveying, flattening, feeding-stacking, reading, and error detecting. To be able to accumulate a stack of banknotes accurately, it is necessary to ensure that every bill is flat with no bent corners. The aim of the present design task is to develop a means of guaranteeing that every banknote is positioned and flattened correctly. We use the design of the flattener module as a case study to demonstrate the use of the CoStorm program.

During the conceptual design phase, as many as possible conceptual ideas on the CCM problem were generated. The ideation process took place while designers interacted and communicated using CoStorm. CoStorm is a web application, where geographically disperse designers could generate, discuss, and summarize ideas. The designers brainstormed ideas and directly logged them into the system, interactively. They made use of their technical knowledge and own field experience and used the graphical representations of CoStorm to trigger new ideas. At any point in time, CoStorm provided designers with an up-to-date map of fresh ideas and concepts, together with basic term relationships that were being built during this stage of the design process. Basically, it was possible to identify and collate many potential solutions. The software program can be customized and tailored according to the requirements of this design problem. From previous research (Zhang *et al.*, 2015) on a project of another module in ATM (deviation rectifier), we obtained a domain ontology, “ATM160602 Ontology”, covering some general knowledge about ATMs and some specific knowledge about rectifiers, as shown in Figure 3.

To easily understand statements, we name researchers at campus as *R1* and *R2*, and engineers in factory as *E1* and *E2*. All of them share the common name, that is, designers. At the beginning, designers logged in CoStorm. *R1*, as an organizer, started a session about the new project of cash-flattener module, filling the topic of this discussion and selecting “ATM160602 Ontology” as the domain ontology. Then designers joined the session and entered the discussion scene, as shown in Figure 5.

Generally, the user interface is separated into three parts.

In the toolbar at the bottom, there are basic operations for this discussion. In the talking area on the left, designers specified the problem or brainstormed their ideas with text. By typing sentences, clicking send button or enter key, everyone in this session would be broadcasted. Thus, designers can easily share their understanding of the problem or innovative ideas. With “#” pairs marking the terms in one message, the main idea could be easily caught by both other designers and CoStorm. As for the term map area, those terms mentioned in messages would be extracted and

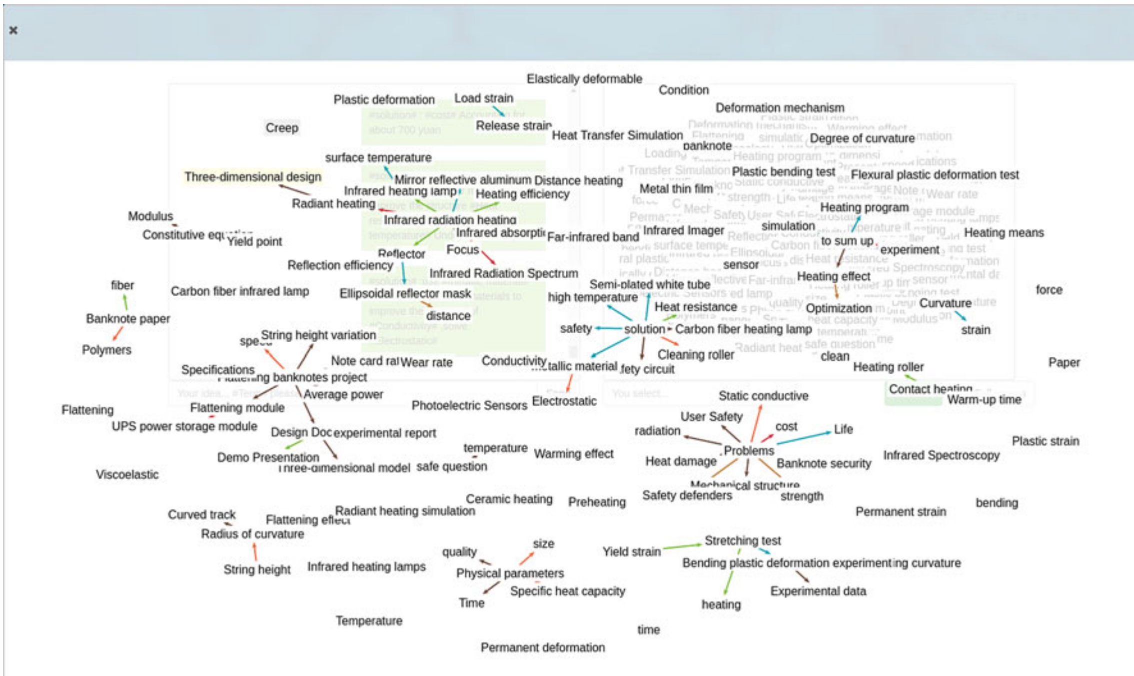

Fig. 6. Term map.

shown. They might even be connected with each other with arrows to show the relations between them. Furthermore, these terms on the map could be dragged and moved for a better view by each designer. Therefore, each of them could focus on what he/she is truly interested in and skilled in.

Furthermore, when this discussion came to a milestone, everything in the design space was saved and designers moved to the succeeding steps, which were reviewing and summarizing the brainstorming.

Test

We applied CoStorm in the practical brainstorm process of designing the above-mentioned flattener module of the ATM. Engineers and researchers from far-away places shared their requirements and ideas within this application.

It should be pointed out that engineers from the industry and researchers from the academe both spoke Chinese. The text log of discussion has been translated into English for this study by an automatic formatter and translator developed by the authors and based on the application of Google Translate. An automation tester has also been developed based on Selenium, a suite of tools for automating web browsers. It has been used to emulate part of the hours of discussion when designers talk about their diverging ideas.

We used the term map, which is an overview of the brainstorming process, revealing how ideas are generated, evolved, branched, and merged. We may check the details of the brainstorm with term map as an interface. Moreover, we use the map to verify if it could lead to some preliminary embodiment configuration.

Results

We collected, overall, 74-sentence or 1353-word chat logs. CoStorm helped to review the entire process that included disordered messages and was useful to clarify the clues and summarize the main points and results. It was possible to pre-process the chat log and replay the brainstorming process using the automation tester and the three strategies mentioned above. In this way, a term map for this discussion was effortlessly built, as shown in Figure 6. It contains 120 statements with 120 terms. Reasoning was produced and organized much better with this map, and designers could readily locate their words and ideas in the discussion.

There is a long path in the map, as shown in Figure 7. This idea flow shows how the problem of flattening a banknote is analyzed. The design problem was to flatten a banknote from a bent shape to a flat shape. To keep the deformation (i.e., banknote flat), the strain should be released. The engineers proposed to keep the shape of the banknote for a period after it had been flattened. They also argued that this process may take a long time according to their experience. Inspired by the idea, the researchers analyzed a number of mechanisms to release strain, and proposed three approaches: by force, by heat, and by humidity. The last one was discarded because water could cause a negative impact on the module. Regarding the use of heat, engineers designed an electric resistance method and carried out some trials. They found that relatively high temperatures were required to achieve good results; thus, this concept was not feasible to implement in the module. The researchers then synthesized a resistance heating system that employed the convection heat-transfer principle. The system designed also triggered the other two energy transfer methods, namely conduction and radiation.

The radiation idea led to a deep discussion, producing a big cluster in the map. It came about when the teammates were talking about "infrared radiation heating", which is an implementation of thermal radiation. According to the map, designers proposed ideas from different perspectives of the topic, and successfully developed them in a much more detailed way. As shown in the term map, an infrared heating lamp could be used as a

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

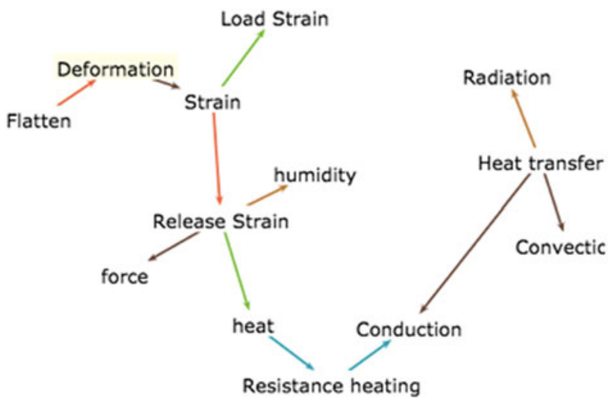

Fig. 7. Term path in the map.

radiation source. An ellipsoidal reflector mask made of reflective aluminum could be equipped for transfer efficiency. The relative positions among lamp, reflector, and banknote must be considered. The map also led to some measurements for the banknote. Engineers measured the infrared radiation absorption spectrum of a banknote, as shown in Figure 8(a), and selected the proper heating lamp. A 3D printed prototype of the heater with an infrared lamp to test the efficiency of heating, as shown in Figure 8(b) and (c), was tested.

Similarly, the design team brainstormed the other four functions of the flattener module, using CoStorm. Combining all the term maps, the design team collected ideas on how to decompose and implement the functions. This led to the embodiment and detail design of the flattener, as shown in Figure 9.

Discussion

The term map is a real-time and an organized summary of what the design teams are producing during the conceptual design phase. The map provides a clear and structured view of what has been discussed, and therefore, it constitutes a valuable tool that supports the ideation stage. Most designers that participated in the CCM design project had the experience of developing systems using the conventional whiteboard and sticky notes brainstorm method. During the use of CoStorm, comparison with the customary method was constantly made. Basically, one of the major concerns was the fact that there was no leader or facilitator during the creative process. It was thought that no useful and feasible outcome could be obtained with the software. At the end of the exercise, however, the team members concluded that CoStorm supported the generation of new ideas and having a graphical representation of the history and evolution of concepts was regarded as very beneficial. The interaction with the maps helped designers with the ideation and decision-making process. For example, when tracking the ideation path, we found that the researchers and engineers are good at finding different ways of synthesizing and analyzing the design problem. The researchers, with solid knowledge backgrounds, often proposed multiple branches based on a single idea. When talking about releasing strain, they easily shifted the force method to heat and humidity methods. The engineers, on the other hand, proposed more concrete ideas. When talking about heat methods, they immediately recalled that they had already tried resistance heating. They also prevented some branches, which were infeasible in terms of manufacturing or practical usage.

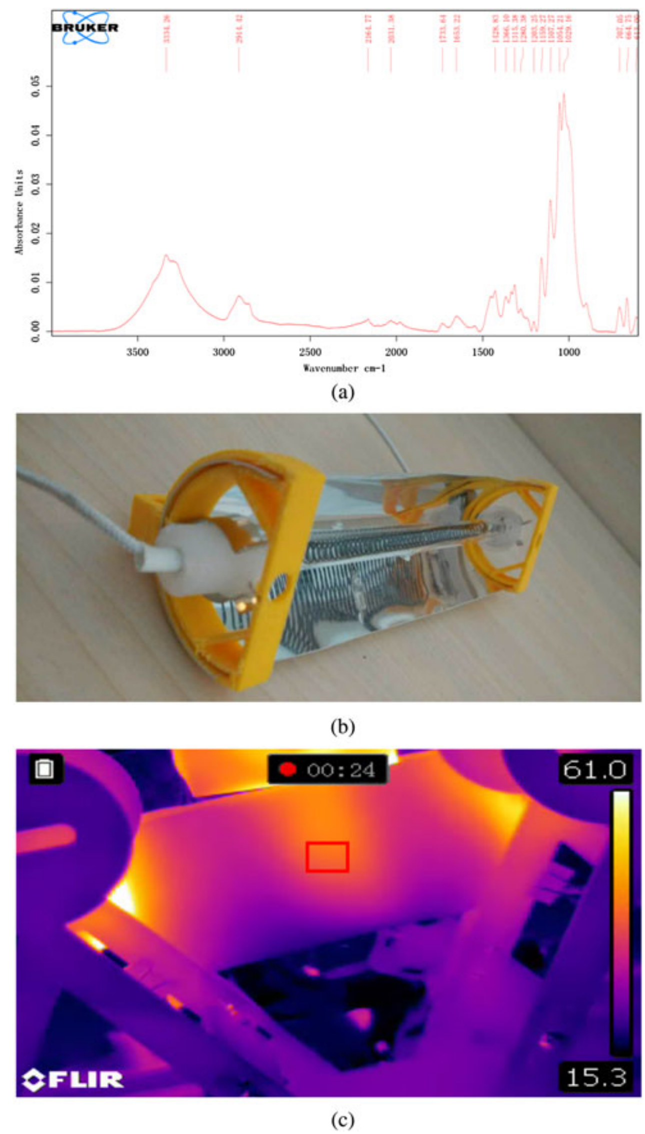

Fig. 8. Implementation of cash-flattener idea. (a) Absorption spectrum of Chinese yuan banknote. (b) Infrared radiation heater prototype. (c) Radiation heating experiment.

The term map showed the reasoning track. The ideas on the track turned to be organized stimuli that inspired new or deeper ideas, thereby integrating designers with difference backgrounds and expertise. Having access to the map produced by CoStorm helped the design team in the process of discussing and deciding over a number of conceptual solutions and alternatives ideas such as using a heating roller, heating press, or a set of electric resistors. Furthermore, the interaction between team members facilitated the analysis on different ways of exchanging thermal energy. Within the cluster of terms, more details were given toward a general idea such as "infrared radiation heating." Those terms, together with the messages behind them, led to the solution of using infrared radiation to flatten the banknote.

We interviewed the CoStorm user designers. One researcher from the university said: "When we got to know the problem at our first meeting, we had a great amount of information and it was not well organised. We tried to use CoStorm and started to get useful guidance from the software. We gradually transformed the problem of note jam into a well-structured task-centred

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

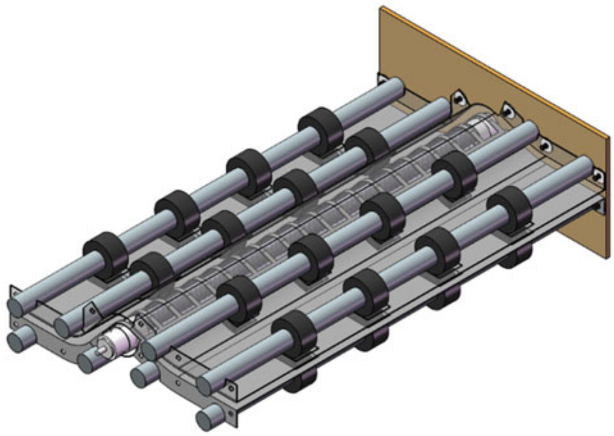

Fig. 9. 3D model of a cash flattener.

Table 1. Case study gains using CoStorm

Advantages	Benefits
Idea generation lead-time	– Construction of a map facilitates visual and abstract ideation process – Team members can log in ideas and concepts any time without interfering with their team's work – Easy to arrange data by topic – Electronic voting is feasible
Overall development time	– Graphical data in real time reduce number of meetings and facilitate decision making – Facilitates electronic meetings – Easier to trace back through the ideation process revealing how and why concepts are originated
Documentation	– Map allows recording ideas, strengths, and weaknesses considered during the design process
Design team engagement	– Team members are free to input ideas and concepts any time with no restrictions; this fact encourages participation – Commitment of every participant is public

design; we focused easily on the major aspects of the flattener module. There were a lot of branches of the design problem, different factors of deformation, and different methods of heat transfer. We thought of a lot of ideas based on our background. They were new and strange to the engineers. When we portrayed them on the map, we knew what we were talking about and also what we had previously discussed as a group. Thus, we did not have to repeatedly explain everything to the engineers. Everything was clearer, and the engineers could contribute to entering fresh ideas to the CoStorm software with their design experience."

These observations match the findings reported in some investigations published so far (Furnham, 2000); computer-aided techniques have helped to overcome shortcomings of conventional methods. Situations such as social loafing (the group context enables individuals to exert less effort), evaluation apprehension (fear of suggesting ideas that might make one look foolish), and production blocking (any group member can suggest an idea at any moment), which are reported by Furnham (2000), were also a matter of discussion during the ATM module project herein. Design team members agreed that conventional brainstorming techniques, although helpful, still presented many of the disadvantages reported in the literature. Table 1 summarizes the benefits of using CoStorm.

Conclusions

In the early stage of a design process, text-based discussion among dispersed or local designers is commonly used for defining a design problem and inspiring preliminary ideas. However, while conversation via instant message goes flatly and linearly, thoughts and ideas are developed and split in unpredictable direction. It is a challenge to make the discussion clearer and bring out a well-organized summary by reviewing the long and plain brainstorming session log. This study proposed models to structure the messages, three strategies for building the term map, and methods to cluster and show the messages on a map. This study helped designers take notes and construct the visualization of ideation process in real time. Divergent and various design ideas are organized in an intuitionistic way, thus inspiring more ideas. The term map also gives a multiple-level view to review the ideas, aiding the decision making in design process. The project reported in this study also successfully developed a prototype software application named CoStorm and applied it to design the flattener module of an ATM. CoStorm proved to be very useful to make the discussion more organized and to inspire designers with new ideas owing to the easy and rapid access to all terms and messages.

Future work comprises the optimization of the algorithm for extracting the relationship between terms and the mechanism of extending ontology with multiple types of relation for a better description of logic. Moreover, according to the characteristic of design states given by a certain designer, his/her profile should be built. His/her design specialty can be analyzed and used to develop a new strategy of cluster design states. Additionally, we are also faced with the challenge of "limited screen space" (Bracewell *et al.*, 2009). We will work on algorithms that can filter out the most significant terms and relevant maps, and provide a multi-level term map. Furthermore, the design states will be expanded to cover sketches. With the extension of design space, this method will be applied to more stages of the design process.

Acknowledgment. This work was supported by the National Natural Science Foundation of China (51175287 and 51675299), China Scholarship Council, and University-Enterprise Cooperation Project with GRG Banking Equipment Co., Ltd.

References

Aurisicchio M, Bracewell R and Armstrong G (2013) The function analysis diagram: intended benefits and coexistence with other functional models. *AI EDAM* **27**, 249–257.

Bracewell R, Wallace K, Moss M and Knott D (2009) Capturing design rationale. *Computer-Aided Design, Computer Support for Conceptual Design* **41**, 173–186.

Cardoso C and Badke-Schaub P (2011) Fixation or inspiration: creative problem solving in design. *The Journal of Creative Behavior* **45**, 77–82.

Chakrabarti A, Sarkar P, Leelavathamma B and Nataraju BS (2005) A functional representation for aiding biomimetic and artificial inspiration of new ideas. *AI EDAM* **19**, 113–132.

https://doi.org/10.1017/S0890060418000215 Published online by Cambridge University Press

Chandrasegaran SK, Ramani K, Sriram RD, Horváth I, Bernard A, Harik RF and Gao W (2013) The evolution, challenges, and future of knowledge representation in product design systems. *Computer-Aided Design, Solid and Physical Modeling 2012* **45**, 204–228.

Chiu S-C and Tomimatsu K (2013) User Involvement in Idea Brainstorming of Design Process: Finding the Effective Strategy in Social Network Service. In *Design, User Experience, and Usability. Design Philosophy, Methods, and Tools* Berlin/Heidelberg, Germany: Springer, pp. 593–598. https://doi.org/10.1007/978-3-642-39229-0_63.

Dennis AR, Minas RK and Bhagwatwar AP (2013) Sparking creativity: improving electronic brainstorming with individual cognitive priming. *Journal of Management Information Systems* **29**, 195–216.

Diehl M and Stroebe W (1987) Productivity loss in brainstorming groups: toward the solution of a riddle. *Journal of Personality and Social Psychology* **53**, 497–509.

Elias EW, Chamakiotis P, Howard TJ, Dekoninck EA and Culley SJ (2011) Can a Virtual Design Environment Enhance Group Creativity and the Use of Stimuli? *ICORD 11: Proceedings of the 3rd International Conference on Research into Design Engineering, Bangalore, India, 10.-12.01.2011*.

Furnham A (2000) The brainstorming myth. *Business Strategy Review* **11**, 21–28.

Gallupe RB and Cooper WH (1993) Brainstorming electronically. *Sloan Management Review* **35**, 27–36.

Gruber TR (1993) A translation approach to portable ontology specifications. *Knowledge Acquisition* **5**, 199–220.

Hoover CW and Jones JB (1991) *Improving Engineering Design: Designing for Competitive Advantage*. Washington, DC: National Research Council.

Howard TJ, Culley S and Dekoninck EA (2011) Reuse of ideas and concepts for creative stimuli in engineering design. *Journal of Engineering Design* **22**, 565–581.

Hsu W and Woon IMY (1998) Current research in the conceptual design of mechanical products. *Computer-Aided Design* **30**, 377–389.

Jiang JJ and Conrath DW (1997) Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. *ArXiv Preprint Cmp-Lg/9709008*. http://arxiv.org/abs/cmp-lg/9709008.

Jin Y and Benami O (2010) Creative patterns and stimulation in conceptual design. *AI EDAM* **24**, 191–209.

Kletke MG, Mackay JM, Barr SH and Jones B (2001) Creativity in the organization: the role of individual creative problem solving and computer support. *International Journal of Human-Computer Studies* **55**, 217–237.

Linsey JS, Wood KL and Markman AB (2008) Increasing Innovation: Presentation and Evaluation of the Wordtree Design-by-Analogy Method, January, 21–32. https://doi.org/10.1115/DETC2008-49317.

MacCrimmon KR and Wagner C (1994) Stimulating ideas through creative software. *Management Science* **40**, 1514–1532.

Moulin C, Kaeri Y, Sugawara K and Abel M-H (2016) Capitalization of remote collaborative brainstorming activities. *Computer Standards & Interfaces*, Special Issue on Information System in Distributed Environment **48**, 217–224.

Murphy J, Fu K, Otto K, Yang M, Jensen D and Wood K (2014) Function based design-by-analogy: a functional vector approach to analogical search. *Journal of Mechanical Design* **136**, 101102.

Nagel JKS, Nagel RL, Stone RB and McAdams DA (2010) Function-based, biologically inspired concept generation. *AI EDAM* **24**, 521–535.

Osborn AF (1953) Applied Imagination. http://psycnet.apa.org/psycinfo/1954-05646-000.

Pahl G, Beitz W, Feldhusen J and Grote K-H (2007) *Engineering Design: A Systematic Approach*. Berlin, Germany: Springer Science & Business Media.

Pal U, Liu Y-C and Chakrabarti A (2014) Evaluating FuncSION: a software for automated synthesis of design solutions for stimulating ideation during mechanical conceptual design. *AI EDAM* **28**, 209–226.

Srinivasan V, Chakrabarti A and Lindemann U (2015) An empirical understanding of Use of internal analogies in conceptual design. *AI EDAM* **29**, 147–160.

Sutton RI and Hargadon A (1996) Brainstorming groups in context: effectiveness in a product design firm. *Administrative Science Quarterly* **41**, 685–718.

Vallecillos J, Criado J, Iribarne L and Padilla N (2014) Dynamic Mashup Interfaces for Information Systems Using Widgets-as-a-Service. In *On the Move to Meaningful Internet Systems: OTM 2014 Workshops*, 438–47. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-45550-0_44.

Weas A and Campbell M (2004) Rediscovering the analysis of interconnected decision areas. *AI EDAM* **18**, 227–243.

West DB (2001) *Introduction to Graph Theory*, vol. **2**. Prentice hall Upper Saddle River. http://math.illinois.edu/~dwest/igt/igtpref.ps.

Youmans RJ (2011) Design fixation in the wild: design environments and their influence on fixation. *The Journal of Creative Behavior* **45**, 101–107.

Youmans RJ and Arciszewski T (2014) Design fixation: classifications and modern methods of prevention. *AI EDAM* **28**, 129–137.

Yuizono T, Xing Q and Furukawa H (2014) Effects of Gamification on Electronic Brainstorming Systems. In *Collaboration Technologies and Social Computing*, 54–61. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-44651-5_5.

Zhang C, Tian L and Wu Y (2015) Ontology-Based Adaptive Object Model for Simulation Physical Parameter Management. In *ASME 2015 International Mechanical Engineering Congress and Exposition*, V011T14A001–V011T14A001. American Society of Mechanical Engineers. http://proceedings.asmedigitalcollection.asme.org/proceeding.aspx?articleid=2501296.

Chengwei Zhang is a PhD candidate in the Department of Mechanical Engineering at Tsinghua University. He is also currently a visiting student researcher in the Department of Mechanical Engineering at University of California, Berkeley. His research focuses on design theory and method, collaborative design, and multi-objective and strategic product design.

Marcelo López-Parra has a doctorate degree from Canfield University, UK. He is a full-time Professor of Mechanical Engineering and an affiliated faculty at Facultad de Ingeniería of Universidad Nacional Autónoma de México, Vehicle Dynamics Group, Design of Machine Systems, Design of Innovative Products. He directs the Automotive Engineering Laboratory. His research interests include vehicle dynamics, design methods, and mechanical vibrations. Dr López-Parra has been awarded the highest level (III) from the Science and Technology National Council in Mexico.

Junyu Chen is a PhD student in the Department of Mechanical Engineering at Tsinghua University. His research interests include interdisciplinary simulation, data mining, and design theory and method.

Ling Tian is currently a professor in the Department of Mechanical Engineering, Tsinghua University. She received her PhD degree in mechanical engineering from Tsinghua University, Beijing, China, in 2002. Her research interests include IC equipment, product life-cycle management, and computer-aided design and engineering.