MixPlore: a digital performance using tangible user interfaces based on cocktail mixology

The International Journal of Arts and Technology (IJART), Vol.4, No.2, pp. 133-154
International Journal of Arts and Technology

Subject Area: Arts and Humanities, Visual Arts and Performing Arts, Computer Science - Computer Science Applications
Publisher: Inderscience Publishers
ISSN: 1754-8853
Scopus Coverage Years: from 2008 to Present

Journal Metrics
Scopus Journal Metrics offer the value of context with their citation measuring tools. The metrics below allow for direct comparison of journals, independent of their subject classification. To learn more, visit: www.journalmetrics.com.

- SJR (Scimago Journal Rank) (2014): 0.167
- IPP (Impact per Publication) (2014): 0.348
- SNIP (Source Normalized Impact per Paper) (2014): 0.714

Compare with other journals

Documents available from
Latest issue: Volume 8, Issue 2 (2016)

- 2015: 10 documents
- 2014: 27 documents
- 2013: 25 documents
- 2012: 16 documents
- 2011: 30 documents
- 2010: 22 documents
- 2009: 22 documents
- 2008: 20 documents

SJR, IPP, and SNIP
SJR = Scimago Journal Rank is weighted by the prestige of a journal, subject field, quality and reputation of the journal have a direct effect on the value of a citation. SJR also normalizes for differences in citation behavior between subject fields.

IPP = Impact per Publication (IPP) measures the ratio of citations per article published in the journal.

SNIP = Source Normalized Impact per Paper measures contextual citation impact by weighting citations based on the total number of citations in a subject field.

Open Access Journals
Journals covered by Scopus are indicated as Open Access if the journal is listed in either the Directory of Open Access Journals (DOAJ) or the Directory of Open Access Scholarly Resources (ROAD).

For questions regarding Open Access Journals please visit the Content info page.

For questions and remarks regarding Open Access Journals please visit the Scopus help files.
Abstract: This paper presents MixPlore, a framework for a live digital performance inspired by cocktail mixology. It aims to maximise the pleasure of mixology, presenting the cocktail as a plentiful art medium through which people can fully enjoy new synesthetic content created through the integration of bartending and musical creation. For this, we created tangible user interfaces utilising cup, tin, glass, muddler, costume display, etc. Making cocktails is followed by a performance including music composition. At the end of every repertoire, the performer presents the resulting ‘sonic cocktails’ to the audience.

Keywords: cocktail, recipe, taste, sound, mixology, mapping, TUI; tangible user interface; live digital performance; art; technology.

1 Introduction

Tasting and eating food are very much intertwined to our being, as they are important elements in human lives. Not only we eat food to survive, but also we eat it for the great pleasure that food gives us with its so many different tastes. Also tasting and eating food greatly influence our social interactions and communication since food brings people together when prepared and eaten together (Kirshenblatt-Gimblett, 1999). Therefore, tasting and eating food play pivotal roles in human lives and cultures.

In the area of art, the word ‘taste’ has been traditionally mentioned in association with art appreciation and artistic styles. However, we never ‘literally’ taste with our mouth or digest with our stomach since artworks are not edible. In modern art, there have been several artists who introduced food (without tasting) to art territories. They considered food as a performance medium dissociated from eating, tasting and nourishing (Kirshenblatt-Gimblett, 1999). In Futurist Cookbook (1932), F.T. Marinetti, the Italian Futurist, asserted the importance of food and cooking as Futurist aesthetics and philosophy, and his Cucina Futurista (Futurist Cuisine) was the first systematic approach to performance-oriented aesthetics of food (Delville, 2007). Since then, artists such as Ann Hamilton, Alicia Rios, Jana Sterbak and others used what they learned from Marinetti’s aesthetics in their happenings.

In music, we found a few musicians who explored the encounter between food and music. Recently, for the occasion of the celebration of the Year of Food, Cuisine and Gastronomy, Sónar 2005, an advanced music and multimedia art festival, joined together to create a party with a special showcase in which sounds, food and kitchen utensils shared the spotlight (Sónar, 2005). The showcase had three interesting performances: The Vegetable Orchestra, Plat Du Jour and Cuisine Concrète + Jordi Vilà. In the
performances, performers used food items as musical instruments. In another occasion, chefs cooked dishes and musicians captured sound samples from the process of cooking to make music.

There are studies in the field of human–computer interaction (HCI) related to food and drink in some different ways. iBar (2007) is a table-top interface that created different interactions between people in the context of a bar setting, which is called 'iBar'. Lover’s Cups presents an example of drinking interfaces which improve social interactions and create diverse personal experiences. There have been other types of research about interactive cookbook systems and interactive kitchens in ubiquitous computing environments. Interactive cookbook systems present new cooking interfaces enabling users to easily search for food recipes and prepare for food (Chi et al., 2007; Ju et al., 2001; Terrenghi et al., 2007). Augmented reality systems with computer vision, display and sensors have been applied to an interactive kitchen (Bonanni et al., 2005). These works do not focus on new artistic experiences but on the introduction of computing to cooking areas. HCI has rarely explored the combination of tasting and eating food to provide new user experiences. There seems to be a few examples of using food and technology in digital performance. Sound Kitchen presents a musical performance where the music is created through chemical reaction (Shiraiwa et al., 2003) This example is not directly related to ‘taste’, however, since it is difficult to make an artwork drinkable. Gamelunch, a sonically augmented dining table, creates an experience of the closed loop among interaction, sound and emotion for effective interactive sound design by the means of physically based sound models (Polotti et al., 2008). It is notable that users in Gamelunch explore contradicting and unexpected sound feedbacks, thus experiencing the importance of environmental sounds in everyday-life acts (Polotti et al., 2008). However, tasting food and using recipes do not play a major role in Gamelunch.

Although sound was not a central part in Futurist Cookbook, Marinetti inspires us to introduce cooking procedures to a digital performance. Kirshenblatt-Gimblett proposed that food and performance converge conceptually in three aspects (Kirshenblatt-Gimblett, 1999). Firstly, to perform is to do: making and serving food associate with preparing and performing the act. Secondly, to perform is to behave, which means social practices related to preparing and eating food. Finally, to perform is to show: when doing and behaving are shown, taste becomes a sensory experience and an occasion for convergence in aesthetic faculty (Kirshenblatt-Gimblett, 1999).

The above-mentioned HCI projects show potential in which HCI techniques can contribute towards creating new experience in a digital performance. Recently, Grimes and Harper proposed a new vision of human-food-interaction in HCI and mentioned that making food is to express creativity (Grimes and Harper, 2008). They point out that smelling, preparing, touching and tasting foods, and even remembering past food experiences can evoke emotional responses. They mentioned potential in various exploring systems that match music with food to provide new user experience (Grimes and Harper, 2008).

Previous artworks (Futurist Cookbook, Sound Kitchen) and studies (Grimes et al., 2008; Kirshenblatt-Gimblett, 1999) motivated us to introduce the sense of taste and the context of cooking to a digital performance. Here, we propose cocktails as new media for a performing art. Firstly, cocktails, in themselves, are synesthetic media; they have recipes, visual elements, sounds and tastes. Secondly, bartending is performative in itself; it is a show with choreographed gestures. Thirdly, we think that the procedure of making
cocktails is analogous to that of making sounds. Figure 1 shows the relationship between cocktail and music. Fourthly, cocktail has the right elements that allow us to formulate it as a digital performance. For instance, a cocktail has a simple recipe with a limited number of ingredients compared to other food items. Finally, it is relatively easy for us to design tangible interfaces from general cocktail-ware.

Therefore, our goals are as follows: to create a high-level artistic pleasure through rich expressions supported by digital technology, and to make further integration of human senses such as tasting, listening and seeing in MixPlore. We explore new aesthetic territories by linking taste to musical creation, in which the taste becomes a primary element that interacts with sound elements. We attempted to design appropriate tangible user interfaces (TUIs) for an edible performance. This paper will show how cocktail mixology, taste, sound and technology can be mixed to produce a new synesthetic digital performance.

Figure 1 The analogies between cocktail and music

2 Methodological approach

Through our literature review, we did not find formalised methodologies for designing a digital performance based on cocktails. Therefore, we had to devise our own methodology for the digital performance. In doing so, we faced three major problems:

1. The cocktail is a rich medium, so it has considerable factors in designing a performance such as what ingredients to use, which cocktail-ware to use, how to choreograph the gesture, etc. We needed to narrow down our scope of considerations.

2. We did not find any criteria for choosing necessary sound components and strategising mappings between gestures and auditory attributes.

3. Without dealing with the problems 1 and 2, it is difficult to devise performance scenarios that help us to implement suitable TUIs, and design sound.

In this respect, we chose the bottom–up approach as our design methodology by observing cocktail recipes. We focused on cocktail recipes because the recipes helped us consistently to create the tastes of cocktails and had a lot of crucial information to analyse bartending gestures and design TUIs to be used.
Figure 2 shows our methodological approach. It has four phases: the recipe analysis, the interactive system design, the sound design and the performance design. In the recipe analysis, we first tried out many different cocktail recipes. From this, we arrived at two possible scenarios: the shaking- and the layering-based scenario. Next, we chose two representative recipes (Mojito for the shaking-based scenario and B-52 for the layering-based one), the specific gestures, cocktail-ware and ingredients in each of the scenarios and recipes. In the interactive system design, we implemented the MixPlore system, and then tested its basic functionality. We acquired criteria for sound components and mappings for sound design from our observation in the recipe analysis and the interactive system design. Then, we concluded that we could have two possible sound engines such as the shaking-based engine and the layering-based one; we implemented them based on the criteria. In the performance design, we designed performance repertoires and the stage setting by composing performance recipes.

**Figure 2** The methodological approach for the framework
3 Recipe analysis

The primary design goal was to build an interactive system for making cocktails and sounds simultaneously, and to propose suitable performance scenarios and contents. We started our research by studying existing cocktail recipes. By studying them, we realised that recipes generally included mixing instructions, ingredients, the quantity of each ingredient and tools to be used. If we could formalise procedures of making cocktails from existing recipes, it would be ideal for designing the entire performance. Also, we needed to find criteria for selecting necessary actions and interfaces to be used in the performance.

3.1 Possible scenarios

Considering a great number of existing recipes, we categorise cocktails into two types: the shaking-based type (scenario #1) and the layering-based type (scenario #2). For example, in the case of Mojito, the bartender mixes mint leaves and other ingredients in a shaker, and shakes them together (Regan, 2003). In the case of B-52, she/he carefully pours liquor with different densities with a small spoon to make several visible layers (Regan, 2003, Wikipedia, 2010).

Three video interviews were conducted with professional bartenders to gather more ideas about bartending for the purpose of our research. In the interviews, we recorded and observed their behaviours, and then found that making cocktails generally involves steps of preparing, mixing, serving, drinking cocktails and other actions.

Now, we have two possible scenarios in this research as seen in Figure 3. In the preparation step, the performer prepares for the performance by setting the table. Generally, the act of preparation is included as a part of performance but sometimes can be done in advance, and thus can be excluded from the actual performing. The mixing step, the core part of performance, has two kinds of actions: shaking and layering. With the act of shaking, the performer mixes up cocktail ingredients and creates sound sources by using the tin interface. Different from shaking, the act of layering enables her/him to make multiple layers in the cocktail and sound by using our glass-laying interface. Then, the resulting sonic cocktail may be served to the audience in the step of serving. Finally, the audience drinks it.

Therefore, this entire procedure can become the process of performance. All performance repertoires of MixPlore followed those two scenarios. We will describe scenarios in Section 5.

Figure 3 Two scenarios based on the procedure of making cocktails
3.2 Study of representative recipes: Mojito and B-52

Our study of these recipes allowed us to set criteria and requirements for the entire MixPore system and performance. However, we focus on interfaces and performing gestures in this part.

We chose Mojito as a representative recipe for scenario #1 since Mojito includes the shaking procedure and basic cocktail-ware that can be applicable for other cocktail recipes. From Table 1, we could determine minimum requirements of interfaces for scenario #1:

1. A general shaker interface that contains ingredients and capture simple shaking gestures.
2. Several cup/glass interfaces that contain ingredients and recognize picking-up and pouring gestures.
3. A muddler interface that enables to mash ingredients and recognize muddling gestures.

The strainer was intentionally excluded since it did not belong to the major interfaces. From the standpoint of gestures, each procedure could be divided into atomic gestures or the combination of a few atomic gestures: picking-up, pouring (tilting), putting-down, moving (up-and-down, front-and-back, left-and-right) and rotating. We discovered common elements in the procedures; the muddling and shaking procedure have common atomic gestures such as moving up-and-down and rotating (Table 1).

On the other hand, B-52 was selected as a representative recipe for scenario #2 since it was one of the most well-known layering cocktails. Layering is usually achieved by slowly pouring liqueurs over the back of a small spoon, so that the liquid falls very gently onto the previously poured liquid layer and rests atop it in a new layer (Regan, 2003). Table 2 provides us with minimum interface requirements for scenario #2:

1. A cup/glass interface that contains ingredients and recognizes picking-up and pouring gestures.
2. A cup/glass interface that senses the liquid level.
3. A glass interface that vividly displays the layering.

B-52 has only one procedure: layering. From the gestural perspective, the act of layering can be deconstructed into pick-up and pouring.

Table 1 The study of Mojito recipe

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Ingredients (qty)</th>
<th>Tool</th>
<th>Atomic gesture or combination of atomic gestures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Muddle</td>
<td>Lime wedges (4)</td>
<td>Muddler and mixing glass</td>
<td>Mash ingredients (up–down or rotate)</td>
</tr>
<tr>
<td></td>
<td>Sugar (3 tsp)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mint leaves (10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Add 1</td>
<td>Rum (2 oz)</td>
<td>Mixing glass</td>
<td>Pick up the rum and pour it, put down</td>
</tr>
<tr>
<td>Shake 3</td>
<td>All</td>
<td>Mixing glass</td>
<td>Move the glass quickly backward and forward, up-and-down or rotate</td>
</tr>
<tr>
<td>Add 4</td>
<td>Crushed ice</td>
<td>Collins glass</td>
<td>Pick up the ice, pour it</td>
</tr>
</tbody>
</table>
Table 1 The study of Mojito recipe (continued)

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Ingredients ( qty)</th>
<th>Tool</th>
<th>Atomic gesture or combination of atomic gestures</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 Strain</td>
<td>All</td>
<td>Strainer and collins glass</td>
<td>Put the strainer in the glass and pour the contents, put it down</td>
</tr>
<tr>
<td>6 Top</td>
<td>Club soda</td>
<td>Collins glass</td>
<td>Pick up the soda and pour it</td>
</tr>
<tr>
<td>7 Add</td>
<td>Garnish</td>
<td>Collins glass</td>
<td>—</td>
</tr>
</tbody>
</table>


Table 2 The observation of B-52 recipe (Wikipedia, 2010)

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Ingredients ( qty)</th>
<th>Tool</th>
<th>Atomic gesture or combination of atomic gestures</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Layering</td>
<td>Kahlua (3/4 oz)</td>
<td>Pousse Café glass, small spoon</td>
<td>Pick up the ingredient and slowly pour it using the small spoon</td>
</tr>
<tr>
<td>2 Layering</td>
<td>Irish cream (3/4 oz)</td>
<td>Pousse Café glass, small spoon</td>
<td>Pick up the ingredient and slowly pour it using the small spoon</td>
</tr>
<tr>
<td>3 Layering</td>
<td>Grand Marnier (3/4 oz)</td>
<td>Pousse Café glass, small spoon</td>
<td>Pick up the ingredient and slowly pour it using the small spoon</td>
</tr>
</tbody>
</table>

Additionally, we realised that the mixing duration is not generally included in the cocktail recipes. Determining the duration is essential for the sound and performance because they are time-based. Thus, it was necessary to carefully adjust the mixing duration to the performance duration in sound and performance design.

4 MixPlore system

In our study of recipes and scenarios, we were inspired to design the MixPlore interfaces by ordinary cocktail-ware, tableware or glassware in order to provide interfaces with good affordance and usability for the performer and audience.

These interfaces should have proper controllability, visibility, mobility, feedback and a waterproof surface to guarantee spectacular effects and their stability. The interfaces should also be easily installed and uninstalled at every stage of the repertoire.

The MixPlore system has a set of TUIs, a network manager and a brain part. The TUIs can capture the bartender's simple gestures. The network manager analyses incoming messages from each TUI and then transfers the messages to the brain part. The brain part facilitates the performer's mixing of cocktails and sounds. All components are connected via a wireless/wired network as shown in Figure 4.

4.1 Interactive system design

4.1.1 Tangible user interfaces

The TUIs consist of a cup, a tin, a muddler, a mat, a glass-layering interface and a costume display.
4.1.1.1 Cup and tin These are for mixing sounds and cocktail ingredients. In the procedure of preparing, cups are used for holding each ingredient and modulating each sound before mixing. The tin acts as a shaker to mix sonic elements and cocktail ingredients in the step of mixing. There are also specially designed cups and tins to which microphones are attached for recording live sounds as additional sound sources.

The cup and tin interface capture the values (the acceleration values of x, y and z axes) of the performer's gesture, transmit the values to the network manager to generate sounds and make visual feedback. As shown in Figure 5a and b, the tin and cup have a sensing component, a control logic unit, a communication module, a visual feedback system (LEDs) and a battery (3.3 V). The sensing component has a three-axis accelerometer and a switch. The accelerometer is used to capture the motion data (the acceleration values) of the performer and measure the slope of the tilted cup and tin. The switch is a trigger for generating a sound and visual feedback. The control logic unit has a micro-controller (Atmega168) for sensing and interfacing. The communication module has a Bluetooth component (Sena ESD100) for the wireless communication.

4.1.1.2 Muddler The muddler mixes and mashes the ingredients (e.g. mint leaves or grapes) and sounds (Figure 5c). The LED display on the upper part of the muddler gives us a lot of visual feedback to add spectacular effects to the performance (e.g. shifting and blinking).

4.1.1.3 Mat The mat with switches is a trigger for starting and ending a sound. The mat with force-sensing resistors (FSRs) measures the weight variation of the cups or tins when placed upon the mat. The weight variation creates sound effects and transmits the message of completing a cocktail to the network manager. The simple LED display is visual feedback embedded in each mat.

4.1.1.4 Glass-layering The glass-layering interface has a liquid level detector and two knobs for sound modulation (Figure 6a). The level detector has a capacitive sensor that detects the liquid level. By adjusting the level and layering liquids of different
alcohol proofs, the performer can create an intriguing change of sound. The two knobs control the tempo and timbre of sound.

4.1.1.5 Costume display  The costume display of the performer, as a wearable media, receives gesture information from the tin and muddler via the Bluetooth communication. The display with a 16 × 16 LED grid responds to the performer’s gestures along with the tin and muddler (Figure 6b).

4.1.2 Network manager

We adopted a 1:N Bluetooth network based on the polling communication model. To implement the network, we used Sena Parani MSP100 (Sena, 2007), a Bluetooth Access Point directly connected to a brain PC and Sena ESD100 Bluetooth modules (Sena, 2007) of each agent: the cup, tin, muddler and costume display.

Figure 5  (a) Tin (95 × 185 mm), (b) cup (85 × 130 mm) and (c) muddler (35 × 330 mm)

Figure 6  (a) Glass-layering interface (50 × 240 mm) and (b) costume display (260 × 180 mm)
4.1.3 Brain

Built in Max/MSP (Cycling 74, 2007), the brain consists of an IO mapper and gesture recognition engine.

The IO mapper refers to pre-defined mappings (Tables 4 and 5) among interfaces, gestures and sounds thus the mapper makes the performance possible. The gesture recognition engine determines gestural types of the performer and transfers the results to the IO mapper (Figure 4).

Based on the mapping, the mapper routes gestural parameters (Table 3) to targets such as the sound engines and the costume display. When using the cup, tin or muddler, the recognition engine receives acceleration parameters to discriminate the performer's gestures. To build the gesture recognition engine, we used Gesture Follower (Gesture Follower, 2007), a HMM-based gesture recognition external for Max/MSP.

**Table 3** The specification of atomic gestures and gestural parameters in interface design

<table>
<thead>
<tr>
<th>Atomic gesture</th>
<th>Gestural parameter</th>
<th>Interface</th>
<th>Relation to Gesture Follower</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick up</td>
<td>Trigger value of switch</td>
<td>Mat</td>
<td>No</td>
</tr>
<tr>
<td>Pour (rotate or tilt)</td>
<td>x, y, z-axes angle values of accelerometer</td>
<td>Tin, cup</td>
<td>Yes</td>
</tr>
<tr>
<td>Put-down</td>
<td>Trigger value of switch</td>
<td>Mat</td>
<td>No</td>
</tr>
<tr>
<td>Move (left–right)</td>
<td>x-axis value of accelerometer</td>
<td>Tin, cup</td>
<td>Yes</td>
</tr>
<tr>
<td>Move (up–down)</td>
<td>y-axis value of accelerometer</td>
<td>Tin, cup</td>
<td>Yes</td>
</tr>
<tr>
<td>Move (front–back)</td>
<td>z-axis value of accelerometer</td>
<td>Tin, cup</td>
<td>Yes</td>
</tr>
<tr>
<td>Rotate</td>
<td>x, y, z-axes angle values of accelerometer</td>
<td>Tin, cup</td>
<td>Yes</td>
</tr>
<tr>
<td>Slowly pour</td>
<td>Value of the liquid level</td>
<td>Glass-layering</td>
<td>No</td>
</tr>
</tbody>
</table>

**Table 4** Mapping between parameters for sound engine 1

<table>
<thead>
<tr>
<th>Component</th>
<th>Sound parameter</th>
<th>Gestural parameter</th>
<th>Interface</th>
<th>Atomic gesture</th>
</tr>
</thead>
<tbody>
<tr>
<td>AM</td>
<td>Trigger</td>
<td>Trigger value of switch</td>
<td>Mat</td>
<td>Pick up/put down</td>
</tr>
<tr>
<td>AM</td>
<td>LFO frequency</td>
<td>z-axis value of accelerometer</td>
<td>Tin, cup</td>
<td>Move (front–back)</td>
</tr>
<tr>
<td>FM</td>
<td>Trigger</td>
<td>Trigger value of switch</td>
<td>Mat</td>
<td>Pick up/put down</td>
</tr>
<tr>
<td>FM</td>
<td>Modulator frequency</td>
<td>x-axis value of accelerometer</td>
<td>Tin, cup</td>
<td>Move (left–right)</td>
</tr>
<tr>
<td>FM</td>
<td>Modulation index</td>
<td>y-axis value of accelerometer</td>
<td>Tin, cup</td>
<td>Move (up–down)</td>
</tr>
<tr>
<td>FM</td>
<td>Carrier frequency</td>
<td>x, y, z-axes angle values of accelerometer</td>
<td>Tin, cup</td>
<td>Rotate</td>
</tr>
<tr>
<td>Granulation</td>
<td>Grain density</td>
<td>y-axis value of accelerometer</td>
<td>Muddler</td>
<td>Move (up–down)</td>
</tr>
<tr>
<td>Granulation</td>
<td>Weight of randomness</td>
<td>(selection order, temporal patterns and amount of delay)</td>
<td>Muddler</td>
<td>Move (up–down)</td>
</tr>
<tr>
<td>Granulation</td>
<td>Grain duration</td>
<td>x, y, z-axes angle values of accelerometer</td>
<td>Muddler</td>
<td>Rotate</td>
</tr>
</tbody>
</table>
Table 4 Mapping between parameters for sound engine 1 (continued)

<table>
<thead>
<tr>
<th>Component</th>
<th>Sound parameter</th>
<th>Gestural parameter</th>
<th>Interface</th>
<th>Atomic gesture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convolution</td>
<td>Randomness amount for selection of the source</td>
<td>y-axis value of accelerometer</td>
<td>Tin</td>
<td>Move (up-down)</td>
</tr>
<tr>
<td>Convolution</td>
<td>Randomness amount for playback speed</td>
<td>z-axis value of accelerometer</td>
<td>Tin</td>
<td>Move (front-back)</td>
</tr>
<tr>
<td>Convolution</td>
<td>Randomness amount for x, y, z-axes angle values of mix ratio (Source: target) accelerometer</td>
<td></td>
<td>Tin</td>
<td>Rotate</td>
</tr>
</tbody>
</table>

Table 5 Mapping between parameters for sound engine 2

<table>
<thead>
<tr>
<th>Component</th>
<th>Sound parameter</th>
<th>Gestural parameter</th>
<th>Interface</th>
<th>Atomic gesture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step sequencer</td>
<td>Create new step</td>
<td>Trigger value of switch</td>
<td>Mat</td>
<td>Pick up</td>
</tr>
<tr>
<td>Step sequencer</td>
<td>Pitch of note</td>
<td>Value of the liquid level</td>
<td>Glass-layering</td>
<td>Slowly pour</td>
</tr>
<tr>
<td>Step sequencer</td>
<td>Playback speed</td>
<td></td>
<td>Knob 1</td>
<td></td>
</tr>
<tr>
<td>FM synthesis</td>
<td>Harmonic mod.</td>
<td></td>
<td>Knob 2</td>
<td></td>
</tr>
</tbody>
</table>

4.2 Atomic gestures and gestural parameters

In MixPloë, we defined several atomic gestures such as picking-up, pouring (tilting), putting-down, moving (up-and-down, front-and-back, left-and-right) and rotating. As seen in Tables 1 and 2, atomic gestures are conceptually basic building blocks to follow the procedures in our scenarios. By technically binding atomic gestures with gestural parameters of interfaces, the Gesture Follower can recognize each atomic gesture. Table 3 illustrates relationships among gestural parameters, atomic gestures, interfaces and Gesture Follower.

5 Sound design

In sound design, it was most important to implement sound engines that satisfy conditions for the two scenarios mentioned in Section 3.1. To achieve this goal, we first collected ideas of sound processing models based on Tables 1 and 2. Next, we built each component for sound processing such as sound modulation and transformation. Then, we made and tested mappings among gestural parameters of interfaces, results of Gesture Follower and sound parameters of components. Finally, we designed two sound engines by combining sound processing models and calibrating the mappings.

5.1 Idea of sound processing based on scenario #1

To represent each ingredient in the cup interfaces, sample sounds and oscillators were selected as sound sources in the preparing procedure of scenario #1. First, we adopted amplitude modulation (AM) and frequency modulation (FM) as fundamental sound processing models since they are the basic synthesis models. Further, we needed sound transformations for greater effects than those of FM and AM at the muddling and the
shaking procedure. Granular synthesis would be the best match for the muddling procedure. In computer music, to granulate means to segment a sound signal into tiny grains (Roads, 2001). Thus, we found out a close analogy between granulation and muddling. For the shaking procedure, we chose convolution as the main sound processing method since we wanted to make more colourful and complex results in mixing two sounds.

5.1.1 Sound engines

This part will describe Table 4 that shows the mapping between gestural parameters and sound parameters.

5.1.1.1 AM/FM Sound sources from oscillators are first processed by the AM or FM component. We used LFO frequency as an AM sound parameter. We applied modulator frequency, carrier frequency and modulator index to FM sound parameters.

5.1.1.2 Granulation A real-time granular synthesis is a sound transformation with various parameters. Grain density and grain duration strongly influence the timbre of a sound. It also has a great effect on recognising how a sound varies in time. Therefore, we mapped grain density and grain duration to mixing gestures. However, it is difficult to make subtle sound effects such as cloud sound by controlling only grain density and grain duration. To achieve such sound effects, we needed more sound factors and their combinations: selection order from the input stream (or direction to read buffer), temporal patterns and the amount of side-chain for delay. Such factors were indirectly controlled by weighted randomness mapped to the up-down gesture of the muddler (Table 4).

5.1.1.3 Convolution This component processes convolution with two input sounds. To obtain various convolution effects, several sample sources were randomly selected by the gestural parameters of the tin interface. The randomness by the tin influences the playback speed and mix ratio (two sources: convolution result). Therefore, increasing randomness will cause more irregularly compounded sounds.

5.1.1.4 Sound engine 1 (SE1) The sound engine for scenario #1 (Figure 7) was written with Max/MSP (Cycling 74', 2007). This engine consists of three components: oscillators, AM/FM and transformations. First, the engine has various types of oscillators such as sinusoids, sawtooth, noise and samples. (The number of oscillators is determined by the number of cups.) Then the engine applies AM or FM to each oscillator when each cup gesture is received. Next, the performer can process the granulation or convolution with sounds generated by oscillators through following steps:

Step 1 she/he pours each ingredient to the tin.

Step 2 she/he mashes ingredients by the muddler, or mixes ingredients by the tin.
Figure 7 The schema of SE1 based on scenario #1

SE1 receives Gesture Follower results (atomic gesture plus information on interfaces used) and gestural parameters from IO mapper; the GF result is used to select a type of components such as AM, FM, granulation and convolution to be used. Gestural parameters, such as the trigger switch and the acceleration, can control sound parameters in the selected component. Additionally, we used some virtual studio technology (VST) plug-ins from Pluggo (Cycling '74, 2007) for more sound effects.

5.2 Idea of sound processing based on scenario #2

A step sequencer and simple FM synthesis were selected as basic sound models for scenario #2. Studying the scenario and recipe of B-52, we first found out that the layering procedure would be analogous to adding a new note to a song. From this, we caught an idea that making a layered cocktail, as the given recipe, was a great match for making music with an arpeggiator (a kind of step sequencer). In this case, we used cups, stirrers, mat interfaces and the glass-layering interface.

5.2.1 Sound engine 2

5.2.1.1 Step sequencer (arpeggiator) We built a simple step sequencer which had the same number of steps as that of ingredients. According to Table 5, picking a cup can create a new step in the sequencer. The performer can control the pitch of note by changing the liquid level. The playback speed can be controlled by the knob 1.

5.2.1.2 FM synthesis A monophonic FM synthesis was coupled with the step sequencer. With the knob 2, the performer can control the harmonic modulation of sound.
5.2.1.3 Sound engine 2 (SE2) The engine was designed to synthesise sound by layering liquid. In Figure 8, the central block represents a note modulation with the step sequencer, which modulates the pitch of the note; each step has one note. Picking-up the first cup starts a new sequence that plays the sequence repeatedly. Simply picking-up the second cup then generates a new step. By increasing/decreasing the level of liquid in the glass, the performer brings the pitch up/down of the selected note in real time. To convert the liquid level into the pitch of a note, the sequence rounds off the pitch to the nearest note based on a chromatic scale. To control the tempo of the sequencer and harmonic modulation, we added two knobs to this engine. Finally, the performer ends the performance with a finish shot glass. Then the resulting sound is processed by the convolution of the sequenced music and the finish sample sound.

5.3 Composition

Based on two scenarios and each coupled sound engine, seven repertoires (recipes) have been composed and performed by the authors. In our composition, it was important to have an experimental spirit. For instance, it was crucial to use recipes that result in creating both good tastes and rich musical expressions. On the other hand, each recipe needed its own performance duration. Therefore, we experimentally composed music for recipes that are 2–18 min long for playing.

Figure 8 The sound schema of SE2 based on scenario #2
6 Performance design

We introduced performance recipe to our performance design. Like the cocktail recipe and music score, the performance recipe is a specification that helps the performer to play a performance and create a sonic cocktail at ease. The recipe includes instruction, the order and flow of the performance, cocktail ingredients, cocktail tools, interfaces and their usages, sound components, performance duration, etc.

Based on our research, we devised two sonic cocktails: Alice in Muscat and Hello Good-bye. We modified Mojito for Alice in Muscat and B-52 for Hello Good-bye. Maintaining the basic forms of scenarios #1 and #2, we changed cocktail recipes and some components of the sounds. This chapter will show two performance recipes and then briefly discusses space design suitable for the performance.

6.1 Performance recipe of Alice in Muscat

Alice in Muscat (Figure 9) is based on scenario #1 and its cocktail recipe is as follows:

- Muscat (suitable amount), blue curaçao (1/2 oz), orange extract (1/2 oz), gin (1/2 oz), ice, Alice's voice (taken from audience), cup interfaces, tin interfaces, muddler interface, duration (about 15 min).

1. Put Muscat grapes in the tin interface and muddle the grapes with the muddler interface. The muddler applies granular synthesis to the sound samples.

2. Put each ingredient in the tin interface in the given order. Such ingredients are pre-mapped to the particular sounds (sinusoids and sample sounds). Or, with the cup interfaces, the performer can record the voice of a selected audience during the performance. The recorded voice could be used as one of the sonic ingredients.

3. Mix all the ingredients into a cocktail by shaking. To prolong the duration of the performance and the music, the performer can partially mix up some ingredients instead of mixing all ingredients simultaneously. Then put another ingredient and mix them again. Repeat this process until the last ingredient is added, and then mix it all up. Here, we suggest 10 min.

4. Gestural parameters are pre-mapped to particular sound properties to create different sound effects accordingly; up-down: ring modulation; front-back: adding an echo effect to the particular sound; right-left: controlling panning of the stereo sound and shaker rotation: filtering the entire sound (e.g. dissolve filter).

5. Pour the mixed cocktail into a glass to generate a resulting sound.

6. Serve the cocktail with the sound to an audience.

6.2 Performance recipe of Hello Good-bye

Hello Good-bye (Figure 10) shows how we integrate the cocktail- and note-layering (arpeggio) that modulate simple elements such as tempo and timbre. It is based on scenario #2. The cocktail recipe is as follows:
Syrup (1 oz), mixed orange extracts (2 oz), cola or coffee liqueur (2 oz), another layered finish shot (1 oz), layering glass interface, mat interfaces, stirrers, cups, duration (about 5 min).

1. *Syrup (first cup)*: pick up the first cup and pour its content into the layering glass interface with a level detector (capacitive sensor). The interface brings the pitch up according to the level of the ingredient.

2. *Mixed orange extract (second cup)*: pick up the second cup and pour its content into the interface. Then freely modulate the tempo and timbre with auxiliary knobs.

3. *Cola or coffee liqueur (third cup)*: pick up the third cup and pour its content into the interface. Then improvise by controlling the knobs as you wish.
A layered cocktail in a small glass (fourth cup): a finish shot. Pick up the last small glass and drop the glass into the interface for layering, and to create the resulting sound.

6.3 Space design

MixPlore attempts to create intimate interactions between performers and audience. Thus, its ideal setting is a small stage, like a cocktail bar (Figure 11). The intimacy provides the audience with many chances to directly participate in the performance. An ideal space for MixPlore performance requires a good sound system and bar facilities such as bar tables, chairs, a refrigerator and a sink for washing glassware.

Figure 11 The space design of P.Art.y 2007 (size: 4 m x 4 m x 1 m)

7 Discussions

The first performance of MixPlore was showcased in People, Art and Technology 2007, an international network performance festival, Seoul, Korea, 2007 (P.Art.y, 2007). We also had two workshops for artists and the general audience after the festival. Before and after the festival, we had meetings with professional bartenders. Their valuable feedback gave us materials to further discuss the structure of the interfaces and the performance.
MixPlore

7.1 Methodological aspects

We adopted the recipe-based approach as mentioned in Section 2. This is an effective approach in building an interactive system because the approach gave us formalised design criteria or methodologies. By applying this approach to our research, we were able to implement a feasible interactive system and successfully gave the live performance twice.

The recipe analysis provided us with a lot of intuitive ideas about how we could relate sound making with cocktail making. The analysis was also useful since it helped us to quickly launch the research and proceed towards our goals. Simultaneously, however, this approach narrowed our focus too much to the recipe and the scenarios, which prevented us from looking for other potential. We realised that there were two more approaches for designing cocktail-based digital performances: the sound-based approach and the overall approach. While the sound-based approach is to design the performance from the perspective of sound, in which the sonic recipe takes a higher priority over the cocktail recipe, the overall approach is to design a performance from the integrated perspective of exploring cocktails and sounds together. We will explore the two approaches for the next step in the future.

7.2 System and interfaces aspects

7.2.1 Cup/tin/muddler

When using the cup, tin and muddler interface, the interfaces work with the gesture recognition engine (Gesture Follower) to discriminate several gestures. For more exact recognition, the engine must detect when the performer starts and finishes her/his gestures (the segmentation problem). To solve this problem, the performer pressed pedals to send the starting and ending point of every gesture to the gesture recognition engine.

7.2.2 Glass-layering

The capacitive sensor is used for the glass-layering interface to measure the height of liquid and reflect the height for sound modulation. Since human fingers have capacitance and it can give noises to the capacitive sensor, the performer must carefully handle the interface. This fact decreases in the usability of the glass layering interface and we will solve this problem in the future.

7.2.3 Latency

The entire latency of our system is about 20–40 ms, which is ignorable for a real-time performance. However, we observed that the costume display had the 400–800 ms, which was not suitable for a musical performance. It was caused by the delay of Bluetooth communication and by that of processing LED patterns in the costume display.

7.2.4 Visibility

We devised the costume display for audience to concentrate on the performer and it was effective as visual feedback. Yet, it was suggested that some visual devices be added to
MixPlore for more visual spectacles. It would be visually improved if a multi-touch tabletop interface is together with this work.

7.2.5 Affordance
Since we borrowed the concept of interface design from real cocktail ware, our interfaces, such as the cup, the tin and the muddler, have good affordance. According to them, performers easily learned how to use our interfaces and readily remembered it.

7.2.6 Space limitation
The limitation has to do with the space setting since it is mainly designed for a digital performance especially in the bar context. For a successful performance, the MixPlore needs a kitchen space equipped at least with a refrigerator and a sink.

7.3 Sound design aspects
Mapping is a central part of sound design in MixPlore, and we have tried to look for meaningful relationships between cocktail- and sound-making. For example, we discovered that even though we use the word ‘mixing’ both in cocktail- and sound-making, the term is used differently in the two contexts. Mixing cocktails involves changing the quality of the given ingredients by mixing and coming up with something qualitatively different. In the context of sound, ‘mixing’ has been traditionally used as a way to arrange two or more sound elements in certain ways without changing the given sound materials. Beginning our research from the metaphor of cocktail, we looked for ways to ‘mix’ sounds in the way that cocktail ingredients are mixed (i.e. the act of mixing that changes the quality and the content of the ingredients, i.e. transformation). We adopted two methods: granulation and convolution in scenario #1.

Granulation is a method of utilising particles called ‘grains’ of sound and organising the grains in certain ways to create a new sound. Convolution is a basic process of transforming given sound sources to a different sound. Finally, we have a concept that granulation process translates the muddling technique, and the convolution method translates the shaking technique in making cocktails (since the performer can break sonic ingredients into many pieces (grains) and mix them up by using the muddler, and s/he also can transform input sound sources to a new sound by shaking them). It is valuable that such analogies act as bridges between cocktail- and sound-making.

7.4 Performance aspects
7.4.1 Performance recipe
We began our research by looking at cocktail recipes as our guide. We saw the need to have a guideline not only in mixing cocktails but also making sounds, as well, and thus devised the idea of performance recipe that orchestrates the entire performance. However, the current performance recipe is in the process of developing and the recipe requires further research to make it more systematised that could act like a score. Therefore, in the future work, it is our goal to establish a more systematic and formalised structure.
MixPlore

7.4.2 Performance repertoires

During the performance with new tangible interfaces, we noticed that the audience had difficulties in following the mechanism of interfaces. Such an incident influenced our performance design. Further, we determined to add the interface demonstration to the performance. As a result, we divided our performance repertoires to two categories: the demonstration- and the art-based repertoire.

7.4.3 Performing duration

The duration is directly related to cocktail tastes. In general, it takes less than 2–3 min to make a cocktail in the bar and it is not enough time to give a performance. Thus the main issue is how we could increase the performance duration up to 10–15 min keeping the original taste. The problem is that the prolonged duration can affect cocktail tastes. For example, the long period of shaking and muddling can make worse the taste. Introducing other performing elements (e.g. recording audience’s voice) to the performance contributed to maintaining the cocktail tastes and extending the performance duration.

7.4.4 Interaction with audience

Early in our design stages, we considered including audience participating in our performance. We planned that the performer invited audience to shake the tin or muddler during the performance, or to drink the created sonic cocktail at the end. We welcomed such possibilities.

8 Conclusions and future works

In this paper, we have described MixPlore, a framework for designing a digital performance with TUIs based on mixing cocktails. MixPlore has shown that cocktail is an attractive medium for new performative expressions by introducing the sense of taste and synesthetic interactions.

We conducted the recipe-based approach paying attention to the aesthetical importance of tasting and eating. Through the approach, we explored the new territory of the novel musical expression since this study combined art and technology with cocktail- and sound-making.

From the recipe analysis, we distilled two scenarios (the shaking- and the layering-based) to design interfaces, sounds and performances. The scenarios made it possible to propose new relationships between bartending and music-creating, and to demonstrate the concept of the edible performance by implementing TUIs. Based on the scenarios, we designed two performance repertoires: ‘Alice in Muscat’ and ‘Hello Good-bye’. Those repertoires contributed to stylising the cocktail-based digital performance by illustrating the mappings among interfaces, gestures, sound components and cocktail ingredients. Further, we specially devised performance recipe as an instructional manual.

In this study, we have mainly focused on relationships between sound and cocktail, but we will concentrate on relationships among image, sound and cocktail. The introduction of image will bring more complexity to the entire performance design. As an initial step, we will modify the current mappings and performance recipe. To enhance
audio–visual expressions, a multi-touch table-top interface should be introduced to the interface design since the table-top interface can control images and sounds simultaneously. Another consideration is to redesign the cup and tin interface by attaching an LCD display and a camera module to the interfaces. The redesigned interfaces would amplify the visual effects during the performance. We hope the next version of MixPlore would be more spectacular by integrating image, sound and cocktail.

References


