

# A User Interface Framework for Kinetic Typography-enabled Messaging Applications

Gregor Möhler, Martin Osen, Heli Harrikari

Sony Corporate Labs Europe  
Hedelfinger Str. 61  
70327 Stuttgart  
Germany  
{moehler, osen, harrikari}@sony.de

## Abstract

Kinetic typography has recently emerged as a technology to enhance text with speech-like expressiveness. In this paper we describe a framework for an effective user interface of a kinetic typography-enabled messaging application. It supports the user creating a dynamic text that is easy to read, has a coherent appearance and reflects the user's communicative intentions. Our prototype uses automated phrasing and accentuation to enhance readability. The overall design is based on a visual framework, which allows for a wide range of distinct emotions and is still visually coherent at the same time. With this framework the user can concentrate on generating a text that is in accordance with his intentions by specifying a suitable animation for the desired emotion and emphasizing words that are important for him.

**Categories & Subject Descriptors:** H.5.2 [Information interfaces and presentation]: User Interfaces – *Natural language*

**General Terms:** Design; Human Factors; Languages

**Keywords:** Kinetic typography; dynamic text; animation; messaging

## INTRODUCTION

When we compare the expressiveness of different modalities we discover the richness of speech over text. Whereas the meaning of a text is derived from the linguistic structure of the sentence, i.e. syntax and semantics of the underlying words, prosody plays an additional factor in the case of speech. It can vary or disambiguate the meaning of the words alone, and, therefore, increases intelligibility. Additionally, it may show the speaker's emotion, his intention and involvement in a conversation. Prosody is mainly characterized by phrasing and accentuation [2].

Various means have been developed to overcome the particularly poor textual representation of speech. A rich set of punctuation markers was introduced into the writing system serving different goals like phrasing or sentence mode. Different font faces like bold or italic are traditionally marking emphasis. Users of electronic text on the other side have developed various concepts to make text more conversation-like. The most prominent example of this kind is probably the emoti-

con. However, even with this enriched set of capabilities text still lacks much of the expressiveness of speech.

Kinetic typography can fill the gap between the two modalities. It has been shown that some of the richness of speech can be ported to the textual domain by displaying text *dynamically* [4,7]. Besides the generation of emotive text, dynamic text can also improve the intelligibility of the text by a distinct temporal structure.

Many of the examples of kinetic typography that we see today are generated offline in a time consuming process. Recently some frameworks have emerged that enable the automatic generation of dynamic text [5,7]. However, there is still a considerable amount of editing work to be done to achieve a suitable and pleasant text passage [1,3].

In this paper we outline a framework that generates dynamic text in which the user is only concentrating on the relevant aspects. The framework includes automated procedures for phrasing and accenting, and a visual framework allowing easy access to an emotional animation library.

## GENERAL USER INTERFACE FRAMEWORK

Let us consider a messaging application on a mobile device as a typical application that might benefit from the additional expressiveness of kinetic typography. The user's goal is to create an output that is easy to read, has a coherent appearance and reflects the user's communicative intentions.

In our framework the message text generation process contains the following steps:

1. The user enters the text.
2. The text is intelligently broken into lines by an automated process to enable easy reading with restricted screen sizes.
3. Based on an evaluation of the input text an initial *word animation scheme* is automatically selected for each sentence. The user may change or vary the selected scheme.
4. Keywords are chosen automatically within the input text and displayed more prominently to enhance readability.

5. The user may emphasize particular words even more by choosing an *emphatic* version of the underlying word animation scheme.
6. The user sends off the text.

As an integral part of the interface an automated *text analysis* process takes care to enhance readability. It analyzes the text and decides about natural places to insert line breaks. It also finds keywords for highlighting. These two aspects correspond to phrasing and accentuation in spoken language, which are to a large extent realized unconsciously by speakers. In the same way users should not need to deal with phrasing and accenting, which is the motivation for the automated process. The next section will explain the text analysis in more detail.

Also, the user should not need to think about how to achieve a coherent, aesthetic output. This is the task of the *visual framework* that defines the overall look but at the same time enables the user to choose from a large variety of animations. Its basic elements are the *word animation schemes*. The visual framework is described in a separate section below.

The user can fully concentrate on generating a dynamic text that fulfills his intentions. An initial word animation scheme is selected automatically based on the textual context, and on an estimation of the user's intention and emotion. More specifically, the selection might be based on a pre-defined input signal such as an emoticon, or on a more subtle evaluation of the wording. An important aspect is that this process is transparent to the user and easy to override. The relevant component is still under development and, hence, not further outlined in this paper.

The animated sentence is immediately shown in a preview. Additionally, a description of the animation scheme is displayed. It might e.g. indicate that the input sentence consists of a *greeting* or is in a *happy emotion*. The user can then choose to change the animation scheme. The ranking that results from the previous estimation step can be used beneficially to guide the selection. A preview of the animation is displayed while the user is choosing among different animation schemes.

The user can additionally emphasize words of his choice. This corresponds to emphatic accents in spoken language, which are fully under control of the speaker. Therefore, the user should be in full control of this functionality. Again a preview of the word animation is shown during the selection process.

## TEXT ANALYSIS

### Phrasing

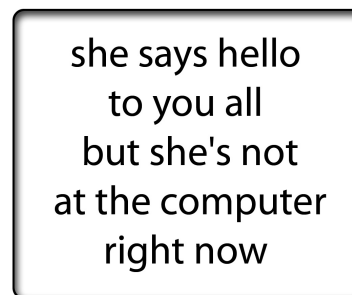
The text entered by the user is automatically divided into lines, similar to any text editor. Since the screen size is often limited in mobile devices, line breaking occurs more frequently than in a normal full-size computer screen. This

decreases the readability of the text. A sentence-breaking algorithm must thus be designed in a way that guarantees the readability of the text even with frequent line breaks. At the same time the original aim for the speech-like expressiveness of textual representations should also be fulfilled. Let us see first how such a sentence breaking is carried out in speech.

Speech can be divided into rhythmic components realized by various prosodic cues for phrase boundaries, such as intonation, duration, and pausing [2]. The basis for the division lies in comprehension: parts forming a unit should occur together. The most common criterion for defining the breaks is the linguistic structure of the sentence, i.e. syntax and/or semantics. For example, the sentence "*she says hello to you all but she's not at the computer right now*" may naturally be divided as follows:

[she says hello] [to you all] but [she's not]  
[at the computer] [right now]

The same prosody-based approach can be carried over to textual representation, where the boundaries indicated by the various prosodic cues in speech are replaced by line breaks (see Figure 1).



**Figure 1. Text with line breaks inserted at prosodically motivated places**

The question now is what happens when the available screen space does not allow a nicely structured phrasing. To overcome this problem, we have developed a solution that simultaneously takes into account both natural linguistically-based phrasing and space limitations ensuring the best possible readability, as is the case with the example in Figure 2.

Furthermore, we assume that linguistic-based phrasing alone does not guarantee good readability, but that also visual considerations are essential. Most importantly the preference for as equal line length as possible, which consequently creates an additional challenge, as the requirement for equal line lengths often results in another outcome than what the linguistic phrasing would prefer.



**Figure 2. Text broken into lines with more severe space limitations**

We have developed a set of rules that define the line breaks on the bases of the words' Part-Of-Speech (POS) tags on the one hand, and the geometry of the word and screen on the other hand. It thus offers a solution for breaking text into lines by taking into account the following three factors:

- Natural, linguistically-driven phrasing based on syntactic/semantic cohesion of the text
- Visual effects: equal line length
- Constraints by small textual spaces

The phrase break algorithm is currently subject of a user evaluation that will shed further light on the relative importance and the interplay of the given factors.

### Accentuation

The second automatic technique for increasing the readability of a text and imitating speech-like properties is accentuation. Similar to speech, where the prominence of essential words is increased sentence-internally by using prosody, also text contains words that should appear more prominent than others. In dynamic text the prosodic cues of speech are transferred into visual ones.

The accentuation tool selects the most important parts of the sentence, i.e. the keywords, which is a useful technique particularly for fast reading. The core idea is to extract individual words that bring essential information to the context. The structure of the sentence (e.g. phrase structure or syntactic roles) plays no role but instead the meaning that individual words carry is considered as the relevant factor, indicated by their POS tags. As an example, nouns and full verbs are highlighted in most cases. The accentuation might be realized in various visual ways, such as bolding. In addition, the non-accented words might be faded out in order to



**Figure 3. Dynamic text with natural line breaking and keyword highlighting**

increase the contrast to keywords, thus making the reading purely keyword-based. The text in Figure 3 demonstrates a dynamic text, as it would appear in a messaging application, with accentuation together with all three criteria of phrasing being exemplified.

### VISUAL FRAMEWORK

In daily life, we are facing animated text quite frequently. Movie titles, TV screen design and commercials use the expressiveness of that medium extensively. Usually, creating such animations requires a lot of time and creative expertise. When designing a system that is generating kinetic text automatically, we have to take into account that a potential user will expect a high level of aesthetic quality.

In order to achieve visually convincing results, we designed a framework, which allows for a wide range of distinct emotions and is still visually coherent at the same time. Our visual framework consists of two layers, namely basic animation schemes and emotional animation schemes.

### Basic Animation Schemes

The first layer defines guidelines for the basic movement of every element (word) in the visual framework. All elements are moving in a similar fashion (e.g. from the bottom of the screen to its center). In a sense, at this stage it is only defined *that* an element is moving. A basic animation scheme alone does not convey any emotional information. It helps to maintain one coherent overall appearance.

A basic animation scheme is not meant to be changed frequently. Still, one of several basic animation schemes can be chosen based on requirements of the communication device, the application or the preference of the user. For instance, for a certain layout, elements coming from the bottom may fit better, while for another one objects coming from the right might be preferable.

### Emotional Animation Schemes

On top of the first layer, the second layer defines any special movements that convey emotion. In other words, at this stage it is defined *how* a certain element is moving (e.g. on it's way from the bottom of the screen to the center).

Emotional animation schemes can occur in two incarnations:

**Subtle animation schemes** typically apply to most words in a sentence. Their purpose is to visualize the long-term mood of a speaker while being subtle enough not to distract the user from reading longer text.

**Emphatic animation schemes** typically apply only to emphatic words actively selected by the user. They represent mainly short-term emotions. Compared with the subtle variant, an emphatic animation is generally much more expressive.

In our system a complete sentence is realized with one emotional category. Every emotional category consists of one subtle and one (or more) emphatic animation schemes. This means, e.g., that a word will be animated differently if the sentence is in an *assertive emotion* or in a *hesitant emotion*. Also, within that sentence, *emphatic* words will show a different behavior than non-emphatic (*subtle*) words.

Additionally, further emotional information can be conveyed by letting single words interact with each other. In general, two basic metaphors have proven to be widely recognized [5]:

**Elements that show personality.** All elements together create a social system where they interact based on social rules (e.g. words start to play or fight with each other, are scared, start to laugh...)

**Elements that show physical behaviors.** All elements together create a physical system where they interact based on physical rules (e.g. words allure each other based on their weight, they collide...)

As mentioned before, in some cases it is beneficial to let the user manipulate certain aspects of the visual appearance. This manipulation is limited to high-level parameters. E.g., the user could control how strong he wants to emphasize a word, but would not have to specify size, speed or other low level parameters. High-level parameters are designed such that they do not scale linearly. A word with an assertive emotion assigned can e.g. make the surrounding words shiver, while an even more assertive one (too assertive, that is) can make them fall apart completely. This makes unexpected and therefore surprising and fun results possible.

### CONCLUSION

In this paper we have presented a framework for an effective user interface of a kinetic typography-enabled messaging application. The text analysis engine is able to determine phrase breaks for varying screen and font sizes. It also finds keywords that can be emphasized visually in the output. Both serve to enhance readability for a broad range of

text types. For future versions of the text analysis we want to look into specific effects that appear in certain text types, such as abbreviations in online chat.

Rhythm, i.e. the pace at which words appear and disappear, is at the current stage only based on the word length. In future we should also consider effects of the underlying emotion. E.g. we can imagine that the rhythm of an *assertive emotion* might be realized differently from a *hesitant emotion*.

It would also be attractive to use our framework for a direct coupling of speech input and dynamic text output such as the one described in [6]. Speech recognition technology as well as automatic phrasing and accentuation extraction from speech would replace the text input and text analysis of our framework. The automated emotion detection of such a system could rely on more features than the text alone, such as the intensity and fundamental frequency of the speech input.

We have developed a visual framework that allows the user to generate an aesthetic and visually coherent output while offering a rich variety of emotional expressions. At the current step we have designed different animation schemes for different emotions and levels of emphasis. It would be beneficial to develop an integrated framework in which the modification of an emotion or emphasis gradually changes the look of the animation.

### ACKNOWLEDGMENTS

We wish to thank Marion Freese for her valuable input to the text analysis engine.

### REFERENCES

1. Bodine, K., Pignol, M. Kinetic Typography-Based Instant Messaging, *Proc. of CHI2003*, ACM Press (2003), 914-915.
2. Clark, J., Yallop, C. An Introduction to Phonetics and Phonology. Blackwell, Oxford (1995).
3. Forlizzi, J., Lee, J.C, Hudson, S.E. The Kinedit System: Affective Messages Using Dynamic Texts, *Proc. of CHI2003*, ACM Press (2003), 377-384.
4. Ishizaki, S. Kinetic Typography: Expressive Writing Beyond the Smileys :-). *Vision Plus Monograph 26 E / D*, International Institute for Information Design (1998), 1-16
5. Lee, J.C, Forlizzi, J., Hudson, S.E. The kinetic typography engine: an extensible system for animating expressive text. *Proc. of UIST2002*, ACM Press (2002), 81-90.
6. Rosenberger, T. Prosodic Font: The space between the spoken and the written. Master thesis, Media Arts and Sciences, MIT (1998)
7. Wong, Y.Y., Temporal Typography – Characterization of time-varying typographic forms. Master's Thesis, Media Arts and Sciences, MIT (1995)