# DeepClouds: Stereoscopic 3D Wordle based on Conical Spirals

**Wolfgang Jentner, Florian Stoffel, Dominik Jäckle, Alexander Gärtner, Daniel Keim**

University of Konstanz
Universitätsstr. 10, 78467 Konstanz

{wolfgang.jentner, florian.stoffel, dominik.jaeckle, alexander.gaertner, daniel.keim}@uni-konstanz.de

**Abstract**

Word clouds are a widely-used technique to visualize documents or collections of documents that are arranged in a space-efficient 2D layout. Their popularity is based on the intuitive understanding and interpretation. 3D computer graphics are available in hand-held devices, on desktop computers, and in the form of specialized hardware, such as Microsofts' HoloLens (Augmented Reality) or the HTC VIVE (Virtual Reality). The wide availability of today's affordable 3D capable devices poses the question, how a 2D word cloud layout can be transferred into 3D space. In this paper, we discuss a prototypical 3D Wordle-based word cloud layout named DeepClouds that generates 3D word cloud layouts by introducing the depth of the position of words as an additional variable in the layout generation algorithm. Hereby, the algorithm exploits the z-buffer to efficiently generate an overlap-free layout from the camera's perspective. Besides introducing the DeepClouds technique, we discuss emerging problems as well as possible future areas of research and applications with respect to 3D word clouds.

**Keywords:** 3D, Word Cloud, Wordle, Augmented Reality, Virtual Reality, Immersive Visualization

## 1. Introduction

Word clouds, such as proposed by Koh et al. (Koh et al., 2010) or Cui et al. (Cui et al., 2010), are a popular choice whenever the main concepts of text-based data collections have to be visualized. They are easy to perceive, to interpret, and have further advantages, such as efficient computability, space efficiency, and typically a visually pleasing appearance, among others. Concept-wise, word clouds are an overlap-free, two-dimensional (2D) arrangement of a set of words, typically ordered by word frequency or an application dependent importance score. The spatial position is determined by a layout algorithm that positions words along a path, such as Archimedean or Euclidean spirals. At the same time, the position is optimized with respect to visual overlap and amount of non-utilized space caused by the 2D word alignment.

Recent advantages in technology enable an affordable access to 3D environments for a wide range of people. Movies are produced to be displayed on modern TVs and cinemas with depth information, leading to a more realistic, immersive experience for the viewer. Lately, mobile devices support techniques called augmented or virtual reality (AR/VR), resulting in many new applications that leverage real-world or artificial environments to present information. Additionally, dedicated hardware for VR and AR is being developed. There exist applications for 3D environments, for example, in the design industry, medical industry, and engineering (Van Krevelen and Poelman, 2010). One commonality is the fact that 3D is mostly used to display 3D information or to create an immersive feeling. An openly discussed research question is whether 3D is useful for information visualization or in other words: are there advantages of presenting abstract information in 3D as opposed to 2D (Butscher et al., 2018). While we cannot give a final answer to this question, we believe that 3D certainly increases the immersive feeling. This is important when dealing with VR and AR technologies and especially of importance for fields such as marketing and advertising. On the other hand, the written language remains our main medium of encoding information and word clouds, besides their drawbacks, remain a popular technique to represent and summarize text content.

A number of different techniques to generate word clouds have been around for some time, but technically their core is similar to what has been already described: an algorithm to place 2D elements on a 2D plane. With recent advances in available processing power and computer graphics technology, we think it is natural to expand the 2D design space of state of the art word clouds to the third dimension, e.g., to include depth in the visualization. In consequence, the word cloud layout has to be generated not only based on the width and height of the visual representation of the words to include, but also by incorporating the third dimension (depth, z-axis). This imposes new problems to solve, for example how the layout-algorithm should compute the overall position, as typically used geometric shapes such as the aforementioned spirals don't take the depth of the 3D space into account. Besides new challenges, there are also new possibilities opening up because of the additional third dimension. The depth, completely unused before, could be used to map another data value in the visual representation of the cloud. Practical use of an additional dimension in the layout can be the addition of another data attribute to be visualized without being interfered by the layout constraints, or to realize streaming word clouds that place recent nearer to the viewer, and older ones fade out after they have been pushed back.

In this paper, we present a first approach of a stereoscopic, 3D word cloud layout called DeepClouds, and elaborate on its technical details. Additionally, we discuss open questions and challenges for advanced 3D layouts, as well as interaction possibilities, which is of interest for applica-

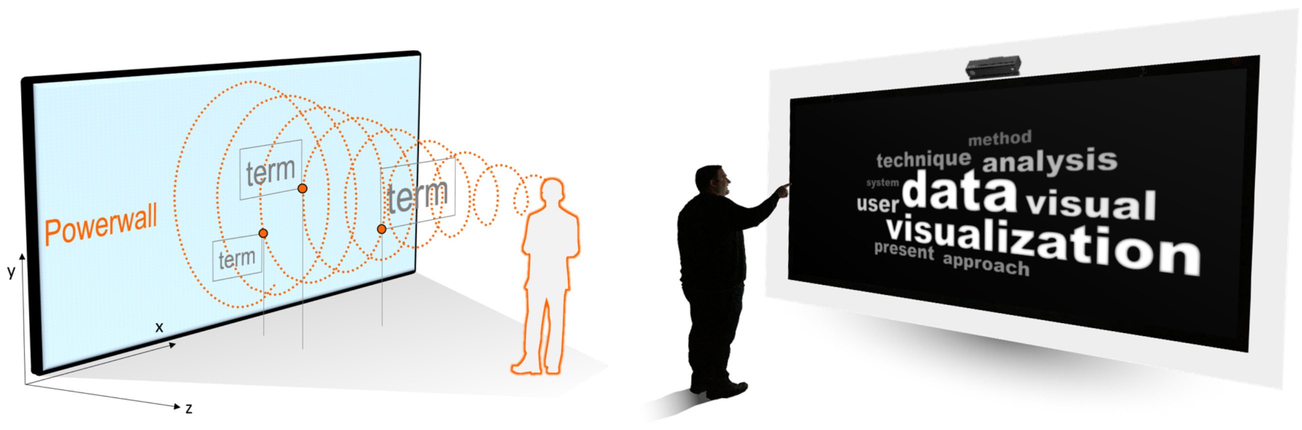


Figure 1: On the left: an illustration of the DeepClouds technique based on conical spirals, exemplifying the computation of the position of words in 3D space. Right: depiction of an envisioned DeepClouds use-case with wall-sized displays and gesture interaction.

tions in immersive 3D environments or large, wall-sized displays, which is the primary environment targeted by DeepClouds.

## 2. Related Work

The origin of tag clouds (also called word clouds) can be dated back far before the computer age (Viégas and Wattenberg, 2008). Flickr, amazon, and others have boosted the popularity of tag clouds tremendously in order to provide an overview of tags and their popularity (Brusilovsky, 1996). Although the tag clouds were discontinued in both cases, the simplicity of tag clouds provides a flat learning curve for users and, typically, no further explanation is necessary to describe the interpretation. Hearst et al. question the usability of tag clouds and suspect that the popularity origins from their visual aesthetics, their popularity among certain design circles, and that word clouds are perceived as being fun, popular, and hip (Hearst and Rosner, 2008). Online services like wordle (Viégas et al., 2009) allow a vast amount of non-expert users to create word cloud visualizations. Word cloud visualizations are also used for information visualization and visual analytics tools to display and summarize documents or corpora (e.g. (Viégas et al., 2007)).

A lot of research has been conducted in developing algorithms that provide space efficient and overlap free layouts, e.g. (Strobelt et al., 2012; Viégas et al., 2009; Seifert et al., 2008)). Other approaches place words according to their semantics, for example given by co-occurrences (Barth et al., 2014), word embeddings (Xu et al., 2016) or common prefixes (Burch et al., 2013). A similar problem setting emerges by animating word clouds. Here, the challenge is to keep the position of the words persistent during animation time (Cui et al., 2010), or during user interactions with the word cloud (Wu et al., 2011). Interactions include the splitting and merging of word clouds whereas the words in the resulting word cloud stay on a similar position. VCloud (Lira et al., 2016) provides the possibility to exclude and join words. Furthermore, it is possible to compare two word clouds of two data sets which is also represented as a word cloud using different colors. ManiWordle (Koh et al., 2010) and WordlePlus (Jo et al., 2015) add interactions to select, add, remove, merge, move, rotate, and resize words. In ManiWordle the user interacts with the word cloud using the mouse. WordlePlus uses pen and touch interactions.

Eventually, other work focuses on the evaluation of word cloud layouts (Lohmann et al., 2009) or the impact of the different visual property mappings (Bateman et al., 2008; Alexander et al., 2017). Rivadeneira et al. identified four main tasks that can be performed with the help of word clouds (Rivadeneira et al., 2007).

Although a lot of research can be found for compact, overlap-free layouts in a 2D space, little amount of work is available for 3D word clouds. WP-Cumulus (Tanck, 2013) is a Wordpress plugin that displays an animated word cloud based on the content of a site. To mimic a 3D effect, words that should be perceived as further in the back are displayed smaller and with a lower opacity. A similar approach is used by JS Tag Sphere (Gork, 2013) where the words are mapped onto a rotating sphere. Itoh et al. use a multi-layer spatio-temporal word-clouds where 2D word clouds are mapped into a space-time cube (Itoh et al., 2016). However, to the best of our knowledge, there is no existing work that presents a word cloud in a (stereoscopic) 3D environment.

## 3. DeepClouds Technique

DeepClouds is inspired by Wordle (Viégas et al., 2009), a well-known visualization technique that arranges words based on their frequency on the display. Thereby, words are positioned overlap-free and space efficient on a 2D canvas. The size and color of the words typically encode the frequency that is the importance of a specific word. We extend Wordle with a third dimension, adding another, unused dimension to the visual variable position. Furthermore, DeepClouds is designed for stereoscopic 3D graphics to emphasize the 3D effect. In the classical Wordle algorithm, the words are successively positioned along an Archimedean spiral. For each word, the algorithm starts at the same origin, and lets the word wander along the spiral until a free spot is found. Free spots on the canvas are determined by testing the bounding box of the word to position with all other, already positioned words, until the bounding boxes do not overlap The main advantage of using a spiral is that unoccupied space is searched for in a radial expanding manner originating at the same reference point, that proves to be an effective search strategy for free space. We apply the very same principle of Wordle in the DeepClouds technique, but additionally encode the importance of a word on the z-axis. This means, the farther away a word is placed, the more dispensable a word is. This brings in two core challenges regarding the positioning of the words and the infinity of space on the z-axis. Following, we discuss both challenges and describe our solutions.

### 3.1. Word Placement

Our placement algorithm for words is based on a hit test between bounding boxes using the z-buffer and stacked conical spirals. The z-buffer is typically a 2D array, each element representing the depth information of one pixel. After an object is rendered, the depth information (z-value) is stored in this special buffer. Analogous to the placement algorithm of Wordle, we perform for each succeeding word a hit test based on its bounding box with all other already placed words. Using the third dimension, a 2D hit test is not enough, which is why we test a hit in the z-buffer: If the bounding box of a word hits another word in the z-buffer, we need to test for a new position in $x$- and $y$-direction but also further back to encode the importance. Performing the hit test in the z-buffer is essential for preserving an overlap-free placement using perspective in a stereoscopic environment.

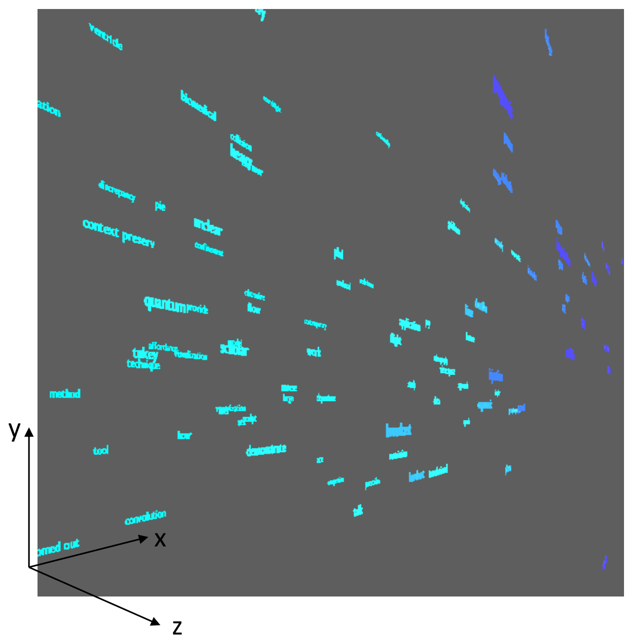


Figure 2: Detail of a DeepClouds word cloud in 3D space shown from the side.

Next, we elaborate on the layout strategy. The overall aim of the layout strategy is to preserve the classical Wordle layout as good as possible while extending it to the third dimension. However, the aforementioned described standard Wordle placement algorithm cannot be applied. You can imagine the problem as follows: The words are ordered according to their frequency, or importance, in advance. Word by word, the algorithm successively determines a free spot without overlap in the canvas. For each word, the algorithm searches for a free spot starting at the origin of the spiral. Following this strategy, a word with lower importance can be placed nearer to the user than a word with higher importance. Summing up, if we would start at the origin for each succeeding word, the concept of depth would not correspond to the importance of a word. Thus, we cannot start from the origin of the spiral for each succeeding word.

In order to preserve the impression of a classical radial Wordle layout and further encode the depth as additional visual variable, we restart the placement algorithm at the z-position of the most recently placed word. This way, we can ensure that less important words are placed further away, however, for the positioning there exist different strategies. Following, we describe three layout strategies depicted in Figure 3 that build on top of each other: a) *Continuous conical spirals.* When the algorithm finds a free spot, it stores the position along the conical spiral ($x$-, $y$-, and $z$-position). For the next word, the search starts at the most recently placed word. The major disadvantage is that the words are spread spaciously on the canvas (see Figure 2). The resulting layout is spacious and does not result in a compact representation that is desired. b) *Continuous stacked conical spirals.* This strategy builds upon the idea of continuing the placement at the position of the most recently placed word. To increase the compactness, word positioning does not continue along the spiral, instead a new conical spiral originating at the most recent position is created. Then, new conical spirals are created recursively until all words are placed. While words are placed closer to each other, there can occur visual artifacts such as the layout spreading into a certain direction. It cannot be guaranteed that the final word cloud hold a shape similar to the spiral. c) *Centered stacked conical spirals.* This strategy follows the concept of recursively originating new spirals at the position of the most recent word. To suit the layout best possible to the classical Wordle layout, we only story the depth value and originate each new conical spiral at the same reference point. All used spirals share the same reference point from which they originate resulting in a Wordle-alike layout.

We argue that the centered stacked conical spirals resemble the classical Wordle layout best in 3D stereoscopic space, as from the presented layout strategies it most closely resembles the spiral shape, as well as is likely to produce layouts with less free space. In the next step, we discuss the concept and usage of a virtually infinite z-axis.

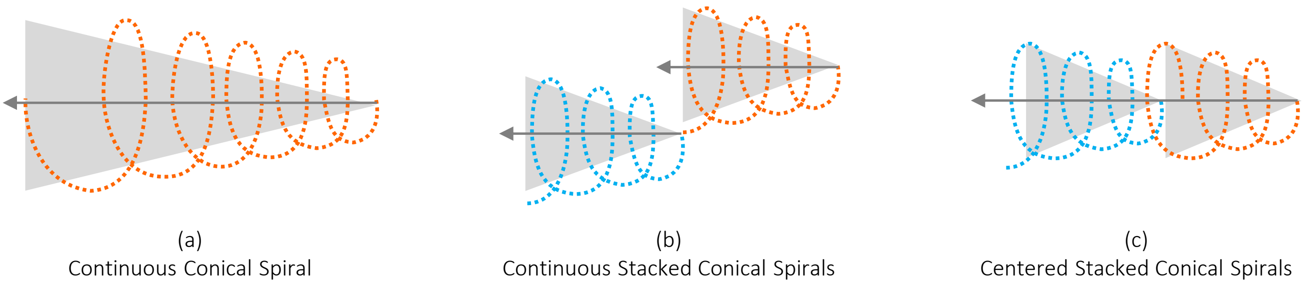


Figure 3: Word placement alternatives in a 3D environment. Using a (a) continuous conical spiral, words spread overall the canvas without minimizing distances between words. (b) Minimizes the distance by starting a new spiral at the position of the most recently placed word at the cost of a skewed overall word cloud shape. The most promising approach to mimic the Wordle layout is (c) that stacks the spirals to assure they share the same origin.

### 3.2. Inverse Zooming

State-of-the-art zooming systems such as maps, mobile applications, and interactive visualizations (e.g. scatter-plots) apply the zooming concept based on the space-scale framework introduced by Furnas and Bederson (Furnas and Bederson, 1995). The authors describe objects as rays leaving the field of view when focusing on specific objects. It goes hand in hand with the concept of geometrically scaling objects; at some point the object is so huge that it can't fit the display anymore. In a stereoscopic 3D environment, we can adapt this concept to the task of navigating through an importance-driven word cloud. In 2D representations, the field of view is restricted by the display dimensions. In contrast, a unique characteristic of 3D is, that the field of view is restricted by the so-called view frustum. The view frustum spans a cropped pyramid into 3D space and removes everything that is not contained in the frustum, from the field of view, which is also known as *view frustum culling*. While moving through 3D space, objects may move within the frustum so that the view is updated automatically. This interaction in 3D space relates to a classical zoom interaction when moving in z-direction.

Because we use the z-axis to encode the importance of words, many words are widely spread further back on the z-axis outside of the frustum. Therefore, we introduce the concept of inverse zooming. The third dimension, that can be zoomed, causes words to literally fly through the viewer position. Compared to classical zooming, we want words to move inside the view frustum towards the center of the field of view, instead of outside. Inverted zooming enables us to infinitely zoom in z-direction and important words to pass us, and are therefore nearer to the viewer, while we focus on less important words, that are farer away. Figure 4 illustrates zoom concepts: (a) Classic zoom, where words move from the center to the display to off-screen on the left, right, bottom, and center. (b) The inverted zoom reverses this behavior and makes words located further back to move towards the center of the field of view. This way, one can successively iterate through all words that are partially visible in the plotted word cloud. Technically, we re-apply the centered stacked conical spirals to all words that surpass the far plane of the view frustum. In other words, once the user zooms the space, words in the center move beyond the virtual viewer position making room for the words that enter the view frustum. In order to fill the opening space, we calculate a new conical spiral at the last known position within the view frustum. The effect we obtain during continuous zooming is that the center of the field of view is iteratively re-filled with less important words, until they become very important, and then surpass us to make room for new words.

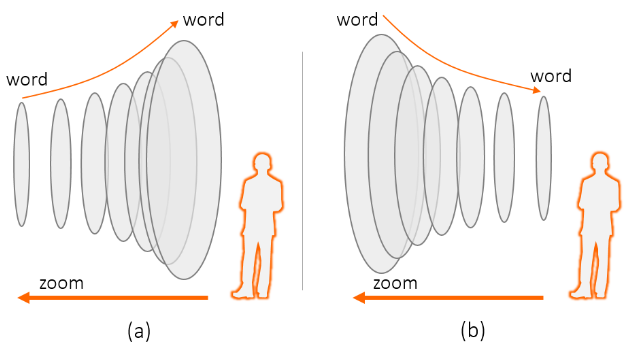


Figure 4: Zoom strategies. (a) state-of-the-art systems: objects in the middle of the focus are magnified and distances between objects increase, objects leave the field of view space while zooming. (b) Inverted zoom: words placed further in the back move into the field of view.

## 4. Discussion and Conclusion

In this paper, we presented a first approach of a word cloud layout algorithm for 3D space. In the following, we discuss some thoughts that arised during design and implementation of a system supporting the DeepClouds technique intended for wall-sized displays with stereoscopic 3D support.

**Overlap and Perspective.** In our view, the most pressing issue of 3D visualization with stacked objects is their perception. Because of that, DeepClouds is designed to produce an overlap-free layout for all possible front-view angles within stereoscopic 3D visualization using the technique described in Section 3.1.. This solves the problem of variable and changing perspective of the viewer caused by the stereoscopic display technology, but at the same time introduces a drawback compared to state of the art 2D wordles: the layout can be described as only *loosely packed*, and lacks space efficiency. While 2D word cloud layout algorithms utilize the available space very efficiently, DeepClouds has to take potentially different view angles as a layout constraint into account, which results in a larger amount of space that is perceived as free. Possibilities to overcome this drawback could be opened up by integrating user interaction and/or user tracking. For example, a refined version of DeepClouds could automatically adjust the positioning of words according to different perspectives caused by the turn of the head or body position of the viewer. This will allow to reduce the free space creation during the layout phase.

**Interaction.** Our current prototype implements selection of words by utilizing a Microsoft Kinect to track the index finger and a pull gesture to select words in the displayed word cloud - or with a modifier on the keyboard apply the inverted zooming concept described in Section 3.2.. After a word has been selected, corresponding documents containing the selected word are shown in an overlay in the upper right part of the screen. We integrated this feature because during the development of the layout algorithm, we recognized that the 3D view seems to make it natural to interact with objects on the screen. Connecting to the previous paragraph about overlap and perspective, gestures to rotate or adjust the representation of the displayed words seem to be a useful addition. Additionally, further operations in the data space besides the described selection of a single word are possible, e.g. to map view space manipulations such as zoom and pan directly to data filters. This is an area where the so called *inverse zooming*, as described in Section 3.2. could be applied and map different operations on the data to the different zooming techniques.

**Virtual Reality.** If the extended interaction possibilities and stereoscopic visualization techniques are combined, the transfer of the resulting technique to a virtual reality environment seems the next logical step. Besides the ability to visualize the word cloud in an immersive 3D environment, the direct interaction with the cloud contents will make the word cloud virtually tangible. As a result, the visualized document collection, as well as user tasks, such as document space exploration or overview, are transferred to a virtual space that can be controlled completely by the developer or designer providing the system. Besides the word cloud visualization, this could mean that the user could be presented with information augmenting the current view in 3D space. This could be beneficial for environments such as libraries or archives to provide a memorable and easily navigable experience for their users, while exploring a potentially large pool of data.

**Augmented Reality.** A sample application for a 3D environment with augmented reality in combination with 3D word clouds could be a content summarization of books in libraries. A user may roam through the shelves and gets information about the books she is looking at. Here, a 3D word cloud combined with a dynamic level of detail could be an effective tool to present the summarized content of the books. In preliminary experiments with Microsofts' HoloLens, we experienced that 2D visualizations appear less immersive whereas even abstract 3D visualizations feel more natural. This is a strong indication that the DeepClouds technique could be useful for AR and VR environments.

We discussed some aspects of the DeepClouds technology and possible realizations, mostly from a technical perspective, as we see this as a first step to the realization of 3D word clouds. Nevertheless, it is important to test our assumptions the resulting design with real users, in order to be able to compare our technique to 2D word clouds, and find areas where our technique needs to be improved, or to have a clear picture of areas where DeepClouds excels the current state of the art of word cloud visualizations. We are optimistic to find such areas, whether they are caused by 3D visualization, new interaction possibilities, or application scenarios that benefit from DeepClouds.

## 5. Bibliographical References


Alexander, E. C., Chang, C.-C., Shimabukuro, M., Franconeri, S., Collins, C., and Gleicher, M. (2017). Perceptual biases in font size as a data encoding. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1.

Barth, L., Kobourov, S. G., and Pupyrev, S. (2014). Experimental comparison of semantic word clouds. In *SEA*, volume 8504 of *Lecture Notes in Computer Science*, pages 247–258. Springer.

Bateman, S., Gutwin, C., and Nacenta, M. A. (2008). Seeing things in the clouds: the effect of visual features on tag cloud selections. In *Hypertext*, pages 193–202. ACM.

Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. *User Model. User-Adapt. Interact.*, 6(2-3):87–129.

Burch, M., Lohmann, S., Pompe, D., and Weiskopf, D. (2013). Prefix tag clouds. In *IV*, pages 45–50. IEEE Computer Society.

Butscher, S., Hubenschmid, S., Müller, J., Fuchs, J., and Reiterer, H. (2018). Clusters, trends, and outliers: How immersive technologies can facilitate the collaborative analysis of multidimensional data. In *2018 ACM SIGCHI Conference on Human Factors in Computing Systems, CHI '18, Montreal, Canada, April 21 - 26, 2018*.

Cui, W., Wu, Y., Liu, S., Wei, F., Zhou, M. X., and Qu, H. (2010). Context-preserving, dynamic word cloud visualization. *IEEE Computer Graphics and Applications*, 30(6):42–53.

Furnas, G. W. and Bederson, B. B. (1995). Space-scale diagrams: Understanding multiscale interfaces. In Irvin R. Katz, et al., editors, *Human Factors in Computing Systems, CHI '95 Conference Proceedings, Den-*

*ver, Colorado, USA, May 7-11, 1995.*, pages 234–241. ACM/Addison-Wesley.

Gork, R. (2013). Js tag sphere.

Hearst, M. A. and Rosner, D. K. (2008). Tag clouds: Data analysis tool or social signaller? In *HICSS*, page 160. IEEE Computer Society.

Itoh, M., Yoshinaga, N., and Toyoda, M. (2016). Wordclouds in the sky: Multi-layer spatio-temporal event visualization from a geo-parsed microblog stream. In *IV*, pages 282–289. IEEE Computer Society.

Jo, J., Lee, B., and Seo, J. (2015). Wordleplus: Expanding wordle's use through natural interaction and animation. *IEEE Computer Graphics and Applications*, 35(6):20–28.

Koh, K., Lee, B., Kim, B. H., and Seo, J. (2010). Maniwordle: Providing flexible control over wordle. *IEEE Trans. Vis. Comput. Graph.*, 16(6):1190–1197.

Lira, W. P., Gama, F., Dall'Agnol, H. M. B., Alves, R., and de Souza, C. R. B. (2016). Vcloud: adding interactiveness to word clouds for knowledge exploration in large unstructured texts. In *SAC*, pages 193–198. ACM.

Lohmann, S., Ziegler, J., and Tetzlaff, L. (2009). Comparison of tag cloud layouts: Task-related performance and visual exploration. In *INTERACT (1)*, volume 5726 of *Lecture Notes in Computer Science*, pages 392–404. Springer.

Rivadeneira, A. W., Gruen, D. M., Muller, M. J., and Millen, D. R. (2007). Getting our head in the clouds: toward evaluation studies of tagclouds. In *CHI*, pages 995–998. ACM.

Seifert, C., Kump, B., Kienreich, W., Granitzer, G., and Granitzer, M. (2008). On the beauty and usability of tag clouds. In *IV*, pages 17–25. IEEE Computer Society.

Strobelt, H., Spicker, M., Stoffel, A., Keim, D. A., and Deussen, O. (2012). Rolled-out wordles: A heuristic method for overlap removal of 2d data representatives. *Comput. Graph. Forum*, 31(3):1135–1144.

Tanck, R. (2013). Wp cumulus.

Van Krevelen, D. and Poelman, R. (2010). A survey of augmented reality technologies, applications and limitations. *International Journal of Virtual Reality*, 9(2):1.

Viégas, F. B. and Wattenberg, M. (2008). Timelines - tag clouds and the case for vernacular visualization. *Interactions*, 15(4):49–52.

Viégas, F. B., Wattenberg, M., van Ham, F., Kriss, J., and McKeon, M. M. (2007). Manyeyes: a site for visualization at internet scale. *IEEE Trans. Vis. Comput. Graph.*, 13(6):1121–1128.

Viégas, F. B., Wattenberg, M., and Feinberg, J. (2009). Participatory visualization with wordle. *IEEE Trans. Vis. Comput. Graph.*, 15(6):1137–1144.

Wu, Y., Provan, T., Wei, F., Liu, S., and Ma, K. (2011). Semantic-preserving word clouds by seam carving. *Comput. Graph. Forum*, 30(3):741–750.

Xu, J., Tao, Y., and Lin, H. (2016). Semantic word cloud generation based on word embeddings. In *PacificVis*, pages 239–243. IEEE Computer Society.