Discourse-driven Comic Generation

Anonymous for Review

No Institute Given

Abstract. Narrative generation enables a range of opportunities for understanding the creative act of storytelling. Prior approaches have mostly converged on a pipeline model, wherein story structure is generated as a precursor to discourse structure, mapping individual story events to discourse elements. This model, however, unnecessarily limits narrative possibilities. We investigate a new generation approach that treats discourse as primary, using *comic generation* as a testbed. Our approach is based on leading discourse theories for comics by McCloud (panel transitions) and Cohn (narrative grammar). Rather than rearranging preexisting panels, we generate panel contents based on notions supported by cognitive theories of visual language. We present a proof-of-concept generator with a wide range of abstract comic output, a computational realization of McCloud's and Cohn's comics theories, and a modular algorithm that affords the evaluation of visual discourse theories.

Keywords: ...

1 Introduction

The computational generation of stories (hereafter *narrative generation*) can help us understand some of the most creative aspects of human intelligence [2], such as reasoning about possible and impossible worlds, and weaving narratives around our daily lives [13]. Historically, narrative generation has followed what Ronfard and Szilas [23] term the *pipeline model*: a narrative artifact is generated by first simulating the story world to form a collection of events, and then piping the event information to a discourse generator, which generates a selective presentation of story world events in a particular medium. A great deal of existing work in narrative generation has primarily pursued this pipeline model [9].

However, as Ronfard and Szilas argue, the pipeline model is neither necessary nor sufficient for narrative generation. Human authors intentionally design their narratives to affect audiences in specific ways, which involves reasoning about how story events are communicated more than which story events occur [1, 3]. It is unnecessary to simulate an aspect of the narrative universe that is never communicated to the audience, if it does not inform the ultimate delivery of the narrative artifact. It is also insufficient to reason about story independent from discourse and medium, as the characteristics of a discourse realization constrain the stories that can be told in that medium [12]. The pipeline model unnecessarily restricts how creative the generator can ultimately be, since story world commitments are not revisited when generating discourse. Further, as will be detailed later, narrative authorship depends on audiences being able to fill in the gaps left open in the consumption of a story [14, 24].

Most prior work that uses the pipeline model implicitly assumes text, or spoken verbal language, as the output generation medium, which allows the pipeline model to avoid some of its limitations by baking medium assumptions into the story model. For example, narrative generators can model updates to internal character state, such as emotion or knowledge change, which can simply be described in text. Communicating those occurrences visually poses a significantly greater challenge. Thus, we propose a simple kind of *visual* narrative as a testbed for discourse generation: wordless comics, as in Figure 1. Comics are a relatively unexplored domain of computational narratology [15], and they present a wide range of expressive opportunities not afforded by text.

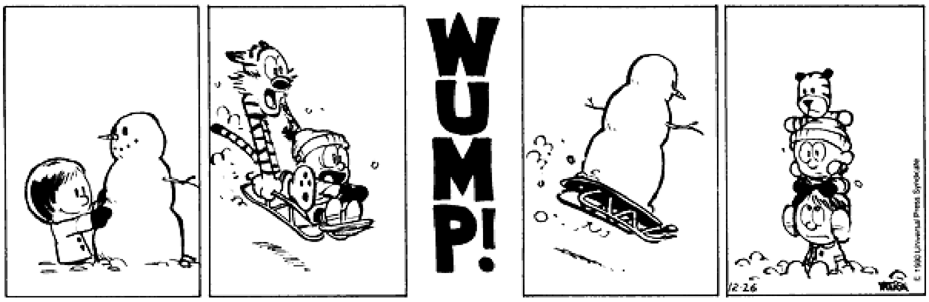

Fig. 1. This wordless Calvin and Hobbes comic strip (© Bill Watterson) exemplifies the kind of output we are targeting with our generator. The strip also illustrates how little plot structure informs this kind of short-form visual storytelling.

Our work represents a departure from the pipeline model, a *discourse-driven* approach to narrative generation, for generating comics. In this model, the story world is only simulated inasmuch as is necessary to support the telling of story events in the discourse; that is, we have a notion of temporal ordering and account for which actants have previously appeared. We present a small-scale computational system [17] to generate comics as a proof-of-concept for our approach.

In the remainder of this paper, we discuss theoretical aspects of comics authoring, our computational implementation of a comic generation system, and our experience with refining our model with linguistic constraints. Our primary takeaway is that both global and local reasoning are important aspects of narrative generation: local reasoning is important for maintaining narrative coherence, and global reasoning is important for maintaining satisfying narrative structure. Both are thus important parts of creating comprehensible comics, and we present an outline of future work designed to explore the human interpretation of our generated artifacts.

2 Generating Comics

Skilled authors convey their stories with knowledge of how information is likely to be processed by an audience. Readers learn to optimize their consumption of relevant information [22], and work to construct inferences [14] about story content in the liminal spaces of discourse (in between sentences in text, panels in comics, scenes in film). Inferences for story content are constructed when they are needed for comprehension and enabled by what has been narrated thus far [19]. The dynamic between story authors and audiences parallels the dynamics of people engaged in cooperative conversation as outlined by the philosopher of language Grice [10]: the storyteller, as the active contributor to the ongoing communicative context, is expected to make her contributions to the discourse based on what is relevant to her narrative intent. These expectations give rise to narrative devices such as *Chekhov's gun*, wherein narrative elements are introduced because they are relevant, and they ultimately demonstrate their relevance at some point in the story. Narrative authors can at the same time flout this expectation of cooperativity in service of a counterpart narrative device, the *red herring*, wherein a story element is introduced and which ultimately has no relevance to the unfolding story.

Purely visual comics, or sequences of visual imagery arranged in panels, present an excellent avenue along which to study discourse theories computationally. The same principles apply: comprehensible comics lack visual clutter, and differences across the *gutters* (gaps between panels) are designed to be filled in by an audience's inference. These principles, as well as notions of brevity, relatedness, and other principles of cooperative narration, manifest in terms of discrete particles that are easily recognized and generated by computer programs.

Comics are structurally similar to written text [24]: they are both made up of individual elements (sentences in text, panels in comics), delimited by special-purpose symbols (full stops in text, panel borders in comics), which can be easily identified, and which can contain a variable amount of information. However, unlike text, comics afford an additional pictorial dimension through which to express information via a palette of visual elements and their spatial relationships to one another, which we might model in terms of their relative size, rotation, horizontal and vertical juxtaposition, and distance. While in general comics offer two dimensions of authorship affordances (textual and visual language), in this work we are concerned only with the pictorial dimension. Saraceni [24] describes three notions of *relatedness* between comic elements, which are the building blocks from which readers may construct meaning inferences. Relatedness, a property of a comic that indicates how its panels are connected or associated, depends on a comic's *cohesion* – the lexico-grammatical features that tie panels together – and *coherence* – the audience's perception of how individual panels contribute to her mental model of the unfolding events. Relatedness emerges from a spectrum of *textual*[1] factors to *cognitive* factors, illustrated in Figure 2. Saraceni distinguishes

[1] *Textual* here does not mean the use of actual text, but rather is a shorthand for *surface code* [28].

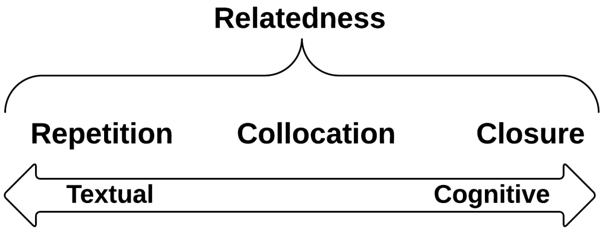

Fig. 2. The spectrum of *relatedness* as discussed by Saraceni [24]. Relatedness indicates how comic panels are connected or associated in the minds of readers, spanning from textual factors to cognitive factors. Along that spectrum, there are three distinguished categories of relatedness: *repetition*, *collocation*, and *closure*, which have demonstrably different effects on the construction of narrative mental models.

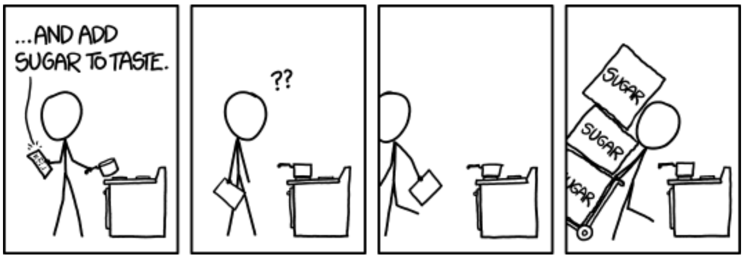

Fig. 3. Strip #1639 of *XKCD*, © Randall Munroe. This comic depends on three aspects of relatedness as described by Saraceni [24], and as illustrated in Figure 2.

three categories of relatedness. Closer to the textual end of the spectrum is the *repetition* of visual elements across panels. Beyond repetition is *collocation*, which refers to an audience's expectation that related visual elements will appear given the ones that have been perceived. Closer to the cognitive end of the spectrum is the *closure* over comic elements, which refers to the way our minds complete narrative material given to us. Closure is terminologically borrowed from the field of visual cognition, but is intended as the mental process of inference that occurs as part of an audience's *search for meaning* [8].

In Figure 3, we see a comic that depends on the three aforementioned aspects of relatedness: first, repetition of the stove and pot is used to maintain cohesion across panels. Second, the punchline of the comic depends on collocation in the sense that we expect "sugar" to come in small measurements, based on non-grammatical domain knowledge. Finally, the comic depends on closure in that we expect the audience to infer several things: that before the start of the comic, the character had been following a recipe; that the character went to retrieve the boxes of sugar between panels 3 and 4; and that the character intends to add sugar to the pot.

In our work we sought to develop a small scale computational model, and thus focused primarily on modeling discourse structure which lies on the textual side of the spectrum. However, our discourse model includes a minimal model of story, which is needed in order to account for some elements of the cognitive side of the spectrum: in particular, we assume chronological ordering between panels and track which visual elements have appeared previously in the panel sequence. We developed two computational models of discourse structure: one based on McCloud's [16]'s account of *transition types*, and the other based on Cohn's [4] theory of visual language.

3 System Description

Our approach to generating visual narratives begins as a linear process that selects next comic panels based on the contents of previous panels, choosing randomly among indistinguishably-valid choices. The concepts we represent formally are *transitions*, *frames*, and *visual elements*, which we define below. There are two levels on which to make sense of these terms: the symbolic level, i.e. the intermediate, human-readable program data structures representing a comic, and the rendered level, designed to be consumed by human visual perception.

A **visual element (VE)** is a unique identifier from an infinite set, each of which is possible to map to a distinct visual representation. We do not explicitly tag visual elements with their roles in the narrative, such as characters, props, or scenery, making the symbolic representation agnostic to which of these narrative interpretations will apply. In the visual rendering, of course, our representation choices will influence readers' interpretation of VEs' narrative roles.

A **frame** is a panel template; at the symbolic level, it includes an identifier or set of tags and a minimum number of required visual elements. The reason a frame specifies a minimum number of VEs is to allow for augmentation of the frame with pre-existing elements: for example, the *monologue* frame requires at least one visual element, indicating a single, central focal point, but other visual elements may be included as bystanding characters or scenery elements. At the rendering level, a frame includes instructions for where in the panel to place supplied VEs. A **panel** is a frame instantiated by specific visual elements.

Finally, a **transition** is a specification for how a panel should be formed as the next panel in a sequence, which we describe formally below.

Transition types were first described by McCloud [16] as a means of analyzing comics. He gave an account of transitions including *moment-to-moment*, *subject-to-subject*, and *aspect-to-aspect*, referring to changes in temporal state, focal subjects, and spatial point-of-view. As Cohn [4] (Chapter 4) points out, these transition types are highly contextual; they presume the audience has a semantic model of the story world in which the comic takes place. For the sake of computational generation, we derived a more *syntactic* notion of transition defined purely in terms of frames and (abstract) visual elements. For example, while McCloud could refer to an action-to-action transition as one where a character is depicted carrying out two distinct actions, we have no notion of *character* and *action*, so instead must refer to which visual elements appear and in which frame. The rendering of a frame itself may position VEs in such a way that an audience would read certain actions or meaning into it; however, this kind of audience interpretation is not modeled to inform generation.

3.1 Formal Transition Types

We introduce six formal transition types: *moment*, *add*, *subtract*, *meanwhile*, and *rendez-vous*, each of which specifies how a next panel should be constructed given the prior sequence.

- **Moment** transitions retain the same set of VEs as the previous panel, changing only the frame.
- **Add** transitions introduce a VE that didn't appear in the previous panel, but might have appeared earlier (or might be completely new). A new frame may be selected.
- **Subtract** transitions remove a VE from the previous panel and potentially choose a new frame.
- **Meanwhile** transitions select a new frame and show *only* VEs that did not appear in the previous panel, potentially generating new VEs.
- **Rendez-vous** transitions select a random subset of previously-appearing VEs and selects a new frame to accommodate them.

3.2 Implementation

Our generator accepts as inputs length constraints (minimum and maximum) and a number of VEs to start with in the first panel. Its output is a sequence of panels (frame names and VE sets) together with a record of the transitions that connect them. The generation algorithm is:

1. Generate transition sequence by choosing transitions uniformly at random, constrained by supplied minimum and maximum length.
2. Generate unique identifiers matching the number of specified starting VEs.
3. Feed transition sequence and starting VEs to the panel sequencer, which selects a next frame and VE set for each new panel based on each transition type's definition (described above). Generate new VEs when necessary, updating the running pool of previously-used VEs at each iteration.

We implemented this algorithm in OCaml and additionally implemented a front-end, a web-based renderer (not linked here for anonymous review). The renderer assigns each frame type to a set of coordinates given by percentage of the vertical and horizontal panel size, and then renders panels by placing visual elements at those coordinates. Visual elements are represented by randomly generated combinations of size, shape (circle or rectangle), and color. An example of the generator's output can be seen in Figure 4.

```
[
  [ { "name": "whisper", "elements": [ 1, 2, 3 ] }, [ "Add" ] ],
  [ { "name": "carry", "elements": [ 4, 1, 2, 3 ] }, [ "RendezVous" ] ],
  [ { "name": "whisper", "elements": [ 4, 1 ] }, [ "Moment" ] ],
  [ { "name": "aid", "elements": [ 4, 1 ] }, [ "End" ] ]
]
```

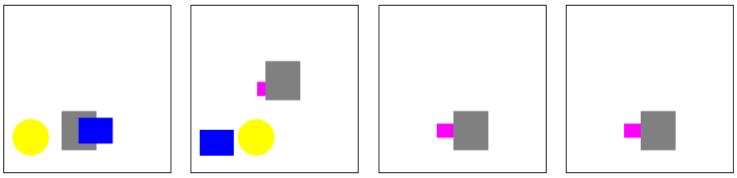

Fig. 4. Example of generator output. While the narrative here is ambigious, we read several things into it: the repetition of the grey (largest) rectangle in every frame suggests it as a focal point, and the sudden appearance of the pink (smallest) rectangle suggests an interloper removing the grey rectangle from its initial context (established by the blue rectangle and yellow circle). Together with the names of the frames (reported in symbolic form above the comic), we can read the sequence as follows: the grey rectangle whispers to the blue rectangle, then is carried off by a pink rectangle, who whispers to the grey rectangle and then aids the grey rectangle.

3.3 Constraining Generation with Cohn Grammars

Generating random transition sequences may result in nonsensical output, such as ending a comic with a *meanwhile* frame in which completely new visual elements are introduced at the end of the comic, but not connected to previous elements; see Figure 5 for an example.

In an attempt to understand the global structure of comic panel sequences, Cohn and his colleagues [6] investigate the *linguistic structure* of visual narratives. They claim that understandable comics follow a grammar that organizes its global structure. Instead of transition types, Cohn's grammar of comics consists of grammatical categories (analogous to nouns, verbs, and so on) indicating the role that each panel plays in the narrative. These categories are **establisher**, **initial**, **prolongation**, **peak**, and **release**, which allow the formation of standard narrative patterns including the Western dramatic arc of *initial – peak – release*.

```
[
  [ { "name": "blank", "elements": [ 1, 2, 3 ] }, [ "Add" ] ],
  [ { "name": "blank", "elements": [ 4, 1, 2, 3 ] }, [ "Subtract" ] ],
  [ { "name": "whisper", "elements": [ 4, 2, 3 ] }, [ "Meanwhile" ] ],
  [ { "name": "posse", "elements": [ 8, 7, 6, 5 ] }, [ "End" ] ]
]
```

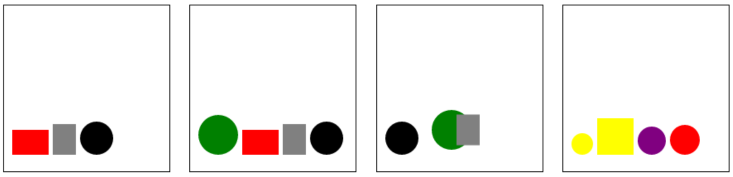

Fig. 5. Example of underconstrained output. The final panel does not maintain relatedness to the preceding sequence.

Formally, Cohn gives the following grammar as a general template for comic "sentences," or well-formed arcs:

(Establisher) – (Initial (Prolongation)) – Peak – (Release)

Symbols in parentheses are optional. In our expression of this grammar (and in several of Cohn's examples), we also assume that prolongations may occur arbitrarily many times in sequence.

Grossman built a generator based purely on Cohn's arc grammar,[2] picking hand-annotated panels for each of an *initial*, *peak*, and *release* slot in the comic. However, this generation scheme does not manipulate the internal structure of the comic panels, allowing for less variability in the output than our scheme with visual elements and frames. Additionally, codifying the syntactic structure of individual panels allows us to characterize *relatedness* between panels as described by Saraceni [24]. In our second iteration of the generator, we combine two approaches to discourse, using *global* Cohn grammars to guide the *local* selection of syntactically-defined transitions.

In particular, we enumerate every possible category bigram in Cohn's grammar, such as *initial to prolongation*, *prolongation to peak*, and so on, and describe sets of transition types that could plausibly model the relationship. This mapping is given below:

Establisher	Initial	{Moment, Subtract, Add, RendezVous}
Establisher	Prolongation	{Moment, Subtract, Add}
Establisher	Peak	{Add, Meanwhile}
Initial	Prolongation	{Moment, Subtract, Add}
Prolongation	Prolongation	{Moment, Subtract, Add}
Prolongation	Peak	{Subtract, Add, RendezVous}
Initial	Peak	{Subtract, Add, Meanwhile, RendezVous}
Peak	Release	{Subtract, Add, RendezVous}

This particular mapping is guided by our intuition rather than any kind of systematic symbolic reasoning—we aim to rule out obvious-seeming syntactic errors, e.g. a *meanwhile* transition at the end of an arc, but other constraints are not so easily expressed. For instance, perhaps a *prolongation* should be realized as a repetition of the previous transition, but this information is not available in the bigram model. In future work, we would like to refine the theoretical grounding of the relationship between transitions and grammatical categories.

With this mapping established, we randomly generate an instance of the arc grammar and populate it with an appropriate set of transitions, after which point we simply hook the transition sequence up to the same panel selector from before. Examples of the constrained generator's output can be found in Figures 6 and 7.

4 Related Work

Montfort et al. [18] developed a blackboard architecture called Slant for textual story generation, which primarily specifies and refines plot structure. However,

[2] http://www.suzigrossman.com/fineart/conceptual/Sunday_Comics_Scrambler

```
{
  "sequence": [ [ "Establisher" ], [ "Initial" ], [ "Peak" ], [ "Release" ] ],
  "comic": [
    [ { "name": "blank", "elements": [ 1, 2, 3, 4 ] }, [ "Moment" ] ],
    [ { "name": "whisper", "elements": [ 1, 2, 3, 4 ] }, [ "Meanwhile" ] ],
    [ { "name": "dialog", "elements": [ 6, 5 ] }, [ "RendezVous" ] ],
    [ { "name": "aid", "elements": [ 4, 3, 1, 2 ] }, [ "End" ] ]
  ]
}
```

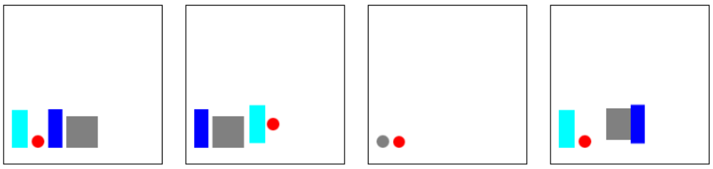

Fig. 6. Example of grammatically-constrained output. This example shows a common pattern in grammatically-constrained output, introducing a new visual element with a Meanwhile transition for the peak, then releasing with a Rendez-vous.

Slant includes a sub-component called Verso, which reasons over narrative discourse as a way to further constrain the narrative plot. Verso detects aspects of the verbs used during the generation of plot structure, and determines the in-progress story's match to a specific genre. Slant is thus not strictly a pipeline model architecture, but the constraints identified during discourse reasoning cannot themselves inform further discourse reasoning. In our approach, discourse reasoning constrains plot structure generation, and can potentially inform further narrative discourse generation.

Pérez y Pérez et al. [20] developed a visual illustrator to the MEXICA system [21], and verified the degree to which their 3-panel comic generator elicited in readers the same sense of story as a textual realization of the same MEXICA-generated plot. While this system still follows the pipeline model of narrative generation, we see their work as complementary: they developed an experiment methodology through which it is possible to empirically assess if their palette of designed visual elements denote story concepts as intended. Future work in discourse-driven comic generation will have to address this point going forward, and Pérez y Pérez et al. provide a step toward understanding the gap between story concepts and the pictorial symbols meant to encode them.

5 Limitations and Future Work

While our interpretation serves as a promising proof-of-concept for concretely interpreting theories of panel relatedness and visual grammar, we have identified a few limitations of our specific implementation choices. Our choice to represent a panel as a frame and, independently, a set of VEs, means that VEs' relationship to the frame, or a VE's role in prior frames, is not available or manipulable. By analogy with textual and verbal language, if a panel is analogous to a sentence,

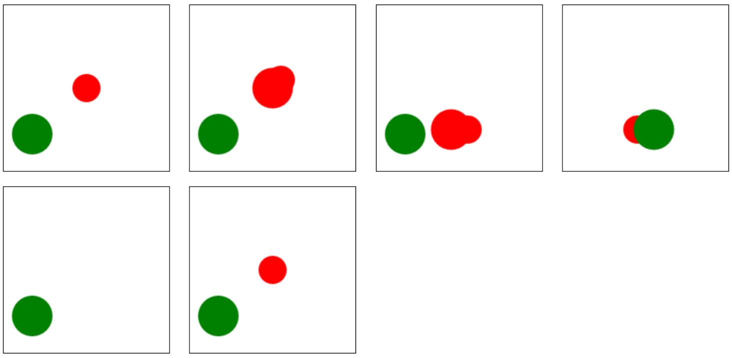

Fig. 7. Example of grammatically-constrained output illustrating a longer sequence with just three visual elements. Potential narrative readings include the green circle throwing the smaller red circle; the larger red circle can be seen as a different entity or as the extension of the smaller one. The peak of this arc is the second-to-last panel.

then we have grammar at the paragraph (narrative arc) level, but not at the sentence level. Also, our choice to generate a transition sequence constrained by a grammar and then feed the transitions to a panel generator (itself a kind of pipeline model) means that the panel generator cannot reflect on the grammatical role of panels to guide its selection. To address these above issues, future work will use linguistic theories to generate panel internals by assigning grammatical roles to VEs that pertain to their visual rendering (such as character, prop, or backdrop), then using those roles consistently across panel sequences. Further, we will seek to reformulate transitions in terms of *edits* on previous panels they are related to, and incorporate theories of semantic scene composition (e.g. [27]) to reason over the grammatical role of panels in sequence.

Another avenue of future work is to empirically evaluate our system. One candidate evaluation involves analyzing the style and variety of our comic generator's output; i.e. our system's *expressive range* [25]. Some metrics based on global properties that are emergent from the point of view of the generator include (a) the number and type of transitions that are generated on average, and (b) the average number of unique readings that an audience comes up with for generated comics. Another candidate evaluation involves analyzing the level of comprehension that our generated comics afford an audience. While there has been work in understanding how people read into narratives involving abstract shapes (e.g. the Heider-Simmel experiment [11]), this evaluation would be more concerned with whether the discourse categories (as discussed by Cohn) that guide the selection of transitions are recognizable by an audience during comprehension. Cohn [5] discusses a methodology through which panel discourse categories can be analytically identified; this analysis would ask whether comic panel categories can be analytically identified by an audience when they are intentionally selected by our generative system.

6 Conclusion

In this work we have presented a discourse-driven approach to narrative generation in contrast to most existing work, which has primarily followed a pipelined approach. We initially designed our system to pay attention to mostly textual factors in comic discourse: the repetition of comic actants across the narrative provides a minimal cohesive backbone on which to pin comic understanding. However, as discussed, this form of generation could generate non-sensical output (e.g. ending comics with a *meanwhile* discourse transition). We therefore appealed to more cognitively-oriented factors via the theory of visual grammar, which helped structure the output in a way that enables other senses of relatedness to contribute to the output's coherence. Thus, through our small-scale system, we have begun to explore the scale and limits of human story sense-making faculties, as well as how they come to bear on narrative generation systems: in our case, through both local and global procedures, which inform cohesion and coherence, respectively. Our algorithms and implementation offer a promising starting point for the computational investigation of discourse-driven narrative.

More broadly, our work highlights the importance of looking to human cognition as a point of departure for the design of narrative generators. Other scholars (e.g. Gervas [9] and Szilas [26]) have argued the same point; our system provides a computational system that demonstrates it. Concretely, the reason for this is that humans bring significant cognitive faculties to bear on the process of narrative comprehension [13]. An instance of this narrative intelligence is our unique ability to fill in the blanks in the liminal spaces of discourse, which (at least) relies on our focalized perspectives into the story world [7]. As our generated comics show, our narrative sense-making abilities allow us to intuit and impose narrative structure on the sequence of depicted images, due to how we fill in the blanks left unspecified in our comics. Therefore, this mental process has a significant role in our appreciation of the narrative artifact, and should have an equally significant role in the generation of it.

References

1. Bordwell, D.: Making Meaning: Inference and Rhetoric in the Interpretation of Cinema. Cambridge: Harvard University Press (1989)
2. Boyd, B.: On the Origin of Stories: Evolution, Cognition, and Fiction. Harvard University Press (2009)
3. Chatman, S.B.: Story and Discourse: Narrative Structure in Fiction and Film. Cornell University Press (1980)
4. Cohn, N.: The Visual Language of Comics: Introduction to the Structure and Cognition of Sequential Images. Bloomsbury, London, England, UK (2013)
5. Cohn, N.: Narrative conjunction's junction function: The interface of narrative grammar and semantics in sequential images. Journal of Pragmatics 88, 105–132 (2015)
6. Cohn, N. (ed.): The Visual Narrative Reader. Bloomsbury Publishing (2016)
7. Genette, G.: Narrative Discourse: An Essay in Method. Cornell University Press (1983)

8. Gerrig, R.J., Bernardo, A.B.I.: Readers as problem-solvers in the experience of suspense. Poetics 22(6), 459–472 (1994)
9. Gervás, P.: Computational Approaches to Storytelling and Creativity. AI Magazine 30(3), 49–62 (2009)
10. Grice, H.P.: Logic and conversation. In: Cole, P., Morgan, J.L. (eds.) Syntax and semantics 3: speech arts. Elsevier (1975)
11. Heider, F., Simmel, M.: An experimental study of apparent behavior. The American Journal of Psychology 57(2), 243–259 (1944)
12. Herman, D.: Toward a Transmedial Narratology. In: Ryan, M.L., Ruppert, J., Bernet, J.W. (eds.) Narrative Across Media: The Languages of Storytelling, pp. 47–75. University of Nebraska Press (2004)
13. Herman, D.: Storytelling and the Sciences of Mind. MIT Press (2013)
14. Magliano, J.P., Kopp, K., Higgs, K., Rapp, D.N.: Filling in the gaps: Memory implications for inferring missing content in graphic narratives. Discourse Processes (2016)
15. Mani, I.: Computational modeling of narrative. Synthesis Lectures on Human Language Technologies 5(3), 1–142 (2012)
16. McCloud, S.: Understanding Comics: The Invisible Art. Harper Collins, New York, NY, USA (1993)
17. Montfort, N., Fedorova, N.: Small-Scale Systems and Computational Creativity. In: Proceedings of the 3rd International Conference on Computational Creativity. pp. 82–86 (2012)
18. Montfort, N., Pérez, R., Harrell, D.F., Campana, A.: Slant: A blackboard system to generate plot, figuration, and narrative discourse aspects of stories. In: Proceedings of the 4th International Conference on Computational Creativity. pp. 168–175 (2013)
19. Myers, J.L., Shinjo, M., Duffy, S.A.: Degree of causal relatedness and memory. Journal of Memory and Language 26(4), 453–465 (1987)
20. Pérez y Pérez, R., Morales, N., Rodríguez, L.: Illustrating a computer generated narrative. In: Proceedings of the 3rd International Conference on Computational Creativity. pp. 103–110 (2012)
21. Pérez y Pérez, R., Sharples, M.: MEXICA: A computer model of a cognitive account of creative writing. Journal of Experimental and Theoretical Artificial Intelligence 13(2), 119–139 (2001)
22. Pirolli, P.: Information Foraging Theory: Adaptive Interaction with Information. Oxford University Press (2007)
23. Ronfard, R., Szilas, N.: Where story and media meet: computer generation of narrative discourse. In: Proceedings of the 5th Workshop on Computational Models of Narrative. pp. 164–176 (2014)
24. Saraceni, M.: Relatedness: Aspects of textual connectivity in comics. In: Cohn, N. (ed.) The Visual Narrative Reader, chap. 5, pp. 115–128. Bloomsbury (2016)
25. Smith, G., Whitehead, J.: Analyzing the Expressive Range of a Level Generator. In: Proceedings of the 2010 Workshop on Procedural Content Generation in Games at the 5th Interational Conference on the Foundations of Digital Games (2010)
26. Szilas, N.: Requirements for Computational Models of Interactive Narrative. AAAI Fall Symposium on Computational Models of Narrative (2010)
27. Zitnick, C., Parikh, D.: Bringing semantics into focus using visual abstraction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3009–3016 (2013)
28. Zwaan, R.A., Radvansky, G.A.: Situation models in language comprehension and memory. Psychological Bulletin 123(2), 162–85 (Mar 1998)