

ビデオコンテンツからの絵コンテコンテンツ生成支援に関する研究

学籍番号：27018606 名前：古川 裕賀 所属：片寄研究室

1. はじめに

従来のレーガンらの研究では、映像コンテンツに対しての感情曲線を、自動的に算出するというものがある。そこで感情曲線の入力、人の手で行った時との差がないかを調査する必要がある。よってメディアコンテンツに対して多数の感情曲線を集める研究が重要であり、その人の手での感情曲線の入力を簡単にするために、映像コンテンツを短くまとめた絵コンテコンテンツが必要である。本研究では、絵コンテコンテンツを「ドラマや映画のストーリーを端的にまとめ、主要な出来事に対応するテキストとそれに対するベストショットの時系列データセット」と定義する。そこで本研究では要約文とその時間情報から、自動的に絵コンテコンテンツを生成するシステムを作ることを目的とする。また、GUI はインタフェース・デザインの勘所[4]を参考にし、手動で絵コンテコンテンツを生成した際よりも時間短縮出来るようにする。

2. システム概要

3.1 システム説明

本研究で作成する絵コンテコンテンツ生成システム(以下「本生成システム」とする)は映像の感情が大きく動く部分の画像を切り取り、要約を乗せてつなぎ合わせるシステムである。絵コンテコンテンツ生成にあたり、入力情報として以下の2つが与えられる。(図1)

①映像の要約文(テキスト)

②その要約文に対応する時間情報(1:20~2:50 など)

映像を切り取る画像の選定方法は感情が大きく動く部分を選定する。感情が大きく動く部分の画像を「ベストショット」と呼ぶことにする。本生成システムは感情が大きく動く部分の画像を数パターン切り取り、本生成システムの利用者にその数パターンの画像の中から最もベストショットであると考えられる画像を選んでもらい、その画像を絵コンテコンテンツに採用する。著作権の観点から、絵コンテコンテンツに使用する画像はOpenCVのノンフォトリアリスティックレンダリング(写実的でない画像を生成する技術の総称)[3]を使用し、画像に加工を施す。映像の切り取り、繋ぎ合わせは映像のカットや編集などが出来るPythonのライブラリであるMoviePy[2]を使用する。

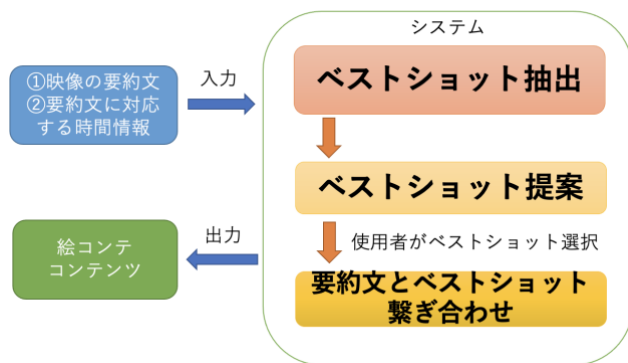


図1: システム概要図

3.2 ベストショット選定方法

本研究では「ドラマや映画の物語に対して人手で設定した主要な出来事に対応する画像として最も相応しいと作業者が選

んだ画像」をベストショットと定義する。筆者らは感情の最も大きく動く部分を、登場人物やキャラクターの表情の感情値が高いシーンであると考えた。既存の要約映像や、感情が大きく動く部分のシーンを切り取った画像等を調査した結果、キャラクターが登場しているシーンであることは前提として、キャラクターの表情に大きく感情が現れているシーンが多くみられた。また、キャラクターの表情の感情が大きく動くということは、大きなイベントが発生している可能性が高いと考察でき、大きなイベントが発生しているシーンは比較的重要であると言える。

表情の感情値の測定方法はGoogle Cloud Vision API [4]を使用する。このAPIの顔検知の感情に関するレスポンスとして、「(喜び)」「(悲しみ)」「(怒り)」「(驚き)」の4項目がある。このそれぞれの項目に対して「(不明)」「(非常に低い)」「(低い)」「(おそらくそう)」「(高い)」「(非常に高い)」という6つの評価がある。本研究では、この評価が高いものを感情が大きく動くと判断し、ベストショットとして提案する。(図2) また池松の研究のベストショット提案手法と組み合わせることでベストショットを提案する。



図2: Google Cloud Vision API スコアの例

3. 評価実験

本生成システムにより絵コンテコンテンツを生成する場合と手動で生成する場合では前者の方が時間の短縮になっていることの有効性を評価するために実験を行う。まずは動画編集ソフトを使って、理想の絵コンテコンテンツを作成する。次に本生成システムを用いて絵コンテコンテンツを生成する。そしてこの2つの手法について5段階で評価してもらい、手動で生成する場合とシステムを使用して生成する場合にどちらの方が時間短縮になっているかを分析する。

4. 現状と今後の予定

現状として、生成システムは完成している。今後は池松のベストショット提案手法と組み合わせた後、表1の通りに評価実験を行う。

表1: 今後の計画表

月	内容
12月	評価実験

参考文献

- [1] MoviePy <https://zulko.github.io/moviepy/>
- [2] Non-Photorealistic Rendering https://docs.opencv.org/4.x/df/dac/group_photo_render.html
- [3] Cloud Vision API <https://cloud.google.com/vision/docs?hl=ja>
- [4] インタフェース・デザインの勘所 https://www.jstage.jst.go.jp/article/jasj/77/4/77_231/pdf