C2VNet: A Deep Learning Framework Towards Comic Strip to Audio-Visual Scene Synthesis

Vaibhavi Gupta, Vinay Detani, Vivek Khokar,
and Chiranjoy Chattopadhyay(✉)

Indian Institute of Technology Jodhpur, Jodhpur, India
{gupta.33,detani.1,khokar.1,chiranjoy}@iitj.ac.in

Abstract. Advances in technology have propelled the growth of methods and methodologies that can create the desired multimedia content. "Automatic image synthesis" is one such instance that has earned immense importance among researchers. In contrast, audio-video scene synthesis, especially from document images, remains challenging and less investigated. To bridge this gap, we propose a novel framework, Comic-to-Video Network (C2VNet), which evolves panel-by-panel in a comic strip and eventually creates a full-length video (with audio) of a digitized or born-digital storybook. This step-by-step video synthesis process enables the creation of a high-resolution video. The proposed work's primary contributions are; (1) a novel end-to-end comic strip to audio-video scene synthesis framework, (2) an improved panel and text balloon segmentation technique, and (3) a dataset of a digitized comic storybook in the English language with complete annotation and binary masks of the text balloon. Qualitative and quantitative experimental results demonstrate the effectiveness of the proposed C2VNet framework for automatic audio-visual scene synthesis.

Keywords: Comic strip · Video synthesis · Deep learning · Segmentation · Multimedia

1 Introduction

Comics are loved widely across the world. People of all age groups read and appreciate them. As digitization increases globally, so are the need to digitize the cultural heritage of which comics beholds a crucial position. It gives a graphical view of society and culture. In the era, people get inclined towards the multimedia aspect of digital documents, so is the case with comics which is a mixture of graphics and texts. The comic book video we see today is being manually created by graphic and video-making tools or software. However, to the best of our knowledge, there has been no work done for the automatic audio-visual synthesis of a comic book. Therefore, we propose a framework Comic-to-Video Network (C2VNet), which evolves panel-by-panel in a comic strip and eventually creates a full-length video (with audio) from a digitized or born-digital storybook. The

J. Lladós et al. (Eds.): ICDAR 2021, LNCS 12822, pp. 160–175, 2021.
https://doi.org/10.1007/978-3-030-86331-9_11

problem statement that we attempt to solve in this paper is "*Given a PDF of a comic book, C2VNet automatically generates a video with enhanced multimedia features like audio, subtitles, and animation*".

Comics consists of different complex elements like panels, speech balloons, narration boxes, characters, and texts. A story or narration emerges from the combination of these elements, which makes reading engaging. Comic research mainly started with reading digital comics on cellular phones [29]. To study the page structure, in [27], a density gradient approach was proposed. In [28], an automated electronic comic adaptation technique was proposed for reading in mobile devices and separating the connected frames. Rigaud [23] simultaneously segmented frames and text area by classifying RoI in 3 categories "noise", "text", and "frame" by contours and clustering. In [18], for manga comics, a three steps approach was proposed. Deep learning frameworks gave a boost to comic image analysis research. A deep learning model was proposed in [13] to find a connection between different panels forming coherent stories. Ogawa et al. [17] proposed a CNN-based model for highly overlapped object problems using anchor boxes.

Speech balloon and narration box segmentation is another area of research in comics analysis. In [10], a deep neural network model was proposed for this task. In [6], extracted the texts in manga comics by following a step-wise process of extracting panels, detecting speech balloons, extracting text blobs with morphological operations, and recognizing text using OCR. In another work, [19] also used OCR for extracting texts and converting text to speech for people with special needs. A morphological approach was proposed in [12].

To do the open and closed comic balloon localization, [22] used active contouring techniques. For text region extraction [7] used two deep learning techniques and classify text regions using SVM. In their further work to recognize typewritten text [24], use OCR Tesseract and ABBYY FineReader and recognize handwritten text. They trained OCR using OCRopus (an OCR system). Later, [21] compared the performances of two systems for pre-trained OCR and other segmentation-free approaches. Many deep learning models using CNN like [30] are widely used to detect text regions from digital documents or images.

Though there has been much work in the analysis of comics elements, comic audio-visual synthesis is less investigated. Comics videos that we see are manually created using graphics and video-making software. The advent of newer technology has changed the habit of viewers' experience and demand. Comics enthusiasts look for new ways to read and explore the comics world. There is the massive popularity of anime, which are Japanese animation based on Manga comics. There is some work in the field of text recognition and converting text to speech. An interactive comic book is created by Andre Bergs [9] named "Protanopia", which has animated images. It is available for android and iOS, which is created using animation software.

To eliminate the manual work, we propose a novel framework, Comic-to-Video Network (C2VNet), which evolves panel-by-panel in a comic strip and ultimately creates a full-length video of a digitized or born-digital storybook. This step-by-step video synthesis process enables the creation of a high-resolution video. The significant contributions of the paper include the following:

Table 1. Details of the number of assets in various categories in IMCDB.

Comic pages	Panels	Text files	Masks	Stories
2,303	9,212	18,424	2,303	331

1. A novel end-to-end comic strip to audio-video scene synthesis framework
2. An improved panel and text balloon segmentation technique
3. A dataset of digitized comic storybooks in the English language with complete annotation and binary masks of the text balloon is proposed.

The rest of the paper is organized as: Sect. 2 presents the details of the proposed dataset. Section 3 presents the proposed C2VNet framework. Section 4 refers to the results and discussions and the paper concludes with Sect. 5.

2 IMCDB: Proposed Indian Mythological Comic Dataset

Machine learning models require training datasets, and applying them to such diverse semistructured comics requires vast datasets. There are several publicly available datasets [1,5,11,13,14]. Indian comics have a wide variety of genres like mythology based on Puranas and The epics; history consists of comics of ancient, medieval, and modern India, Literature, which has comics of folktales, fables, legends, and lore including animal stories and humorous comics. Since there is no Indian Comic dataset available for comic analysis, we have created our dataset IMCDB (Indian Mythological Comic Database) by collecting comic books from archiving websites where these comics are digitally available. We have made ground truth annotations for each panel in pages and ground truth text files for each narration box and speech balloon within a panel. Additionally, ground truth binary masks of speech balloons and narration box for each page. The dataset is available at [2].

IMCDB consists of comics of “Amar Chitra Katha” and “Tinkle Digest”. Amar Chitra Katha was founded in 1967 to visualize the Indian culture of great Indian epics, mythology, history, and folktales. Tinkle Digest also contains comics with famous characters like “Suppandi” and “Shikari shambhu”, which gain popularity among all age groups.

Comics consists of different complex elements like panels, speech balloons, narration boxes, characters, and texts. A story or narration emerges from the combination of these elements, which makes reading engaging. Different comic book elements in the proposed dataset presented in Fig. 1. There are four specific elements of comic books in the dataset which are comic images, panels, speech balloon masks and text files. All the comics in IMCDB are written in the English language, so the reading order is from left to right, hence the order of panels, speech balloons, and texts. Table 1 summarizes the contents of IMCDB dataset.

Comic Images: It contains the comic book PDF pages in “.png” format.

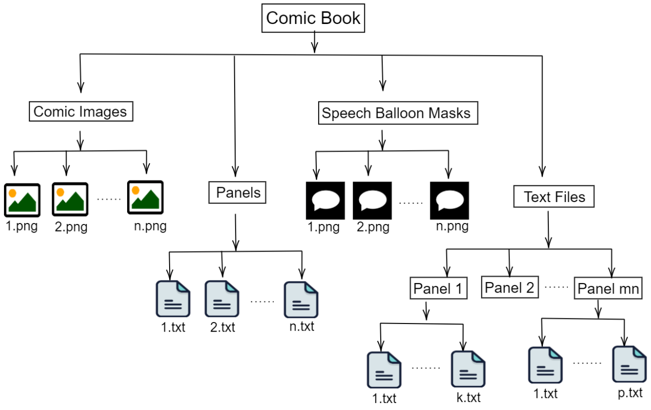


Fig. 1. An illustration of the hierarchy of different assets in IMCDB.

Panels: For ground truths of the panel, graphical image annotation tool "LabelImg" was utilized to create YOLOv3 compatible files. The panels description in "Yolov3" format for each comic page is present in ".txt" files.

Speech Balloon Masks: The speech balloons in IMCDB are of varying shape and size, as well as there are overlaps. We created binary masks as ground truth for all the speech balloons and narration boxes. For each comic book page there is a speech balloon masks in ".png" format.

Texts: The source text is all handwritten in uppercase with varying styles. For ground truth, we provided ".txt" files corresponding to each narration box/speech balloon within a panel of a comic book page.

3 C2VNet Architecture

This section presents the step-by-step approach followed in C2VNet, shown by Fig. 2, to develop interactive comic visuals that evolve panel-by-panel in a comic strip and ultimately creates a full-length video of a digitized or born-digital storybook. It involves four main steps; (1) Preprocessing, (2) Extraction of comic elements, (3) Video Asset creation, and (4) Video composition and animation. In the following sub-section, we describe each framework module in detail.

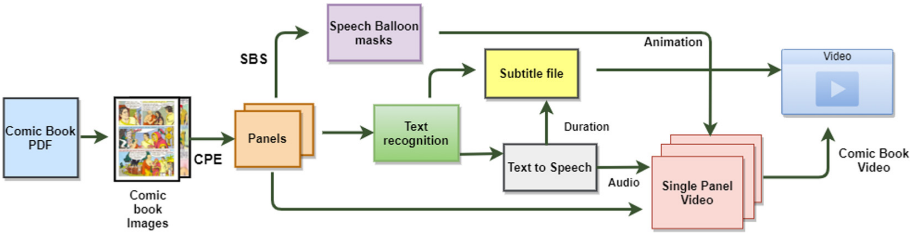


Fig. 2. A block diagram of the proposed C2VNet framework.

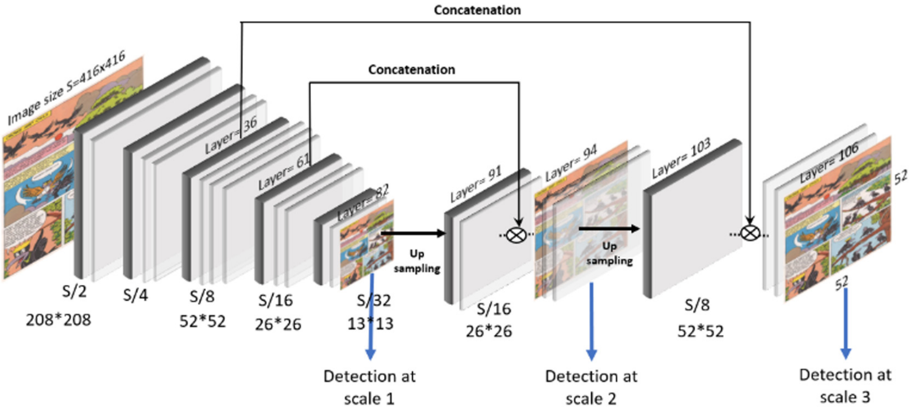


Fig. 3. An illustration of the proposed CPENet model for panel extraction.

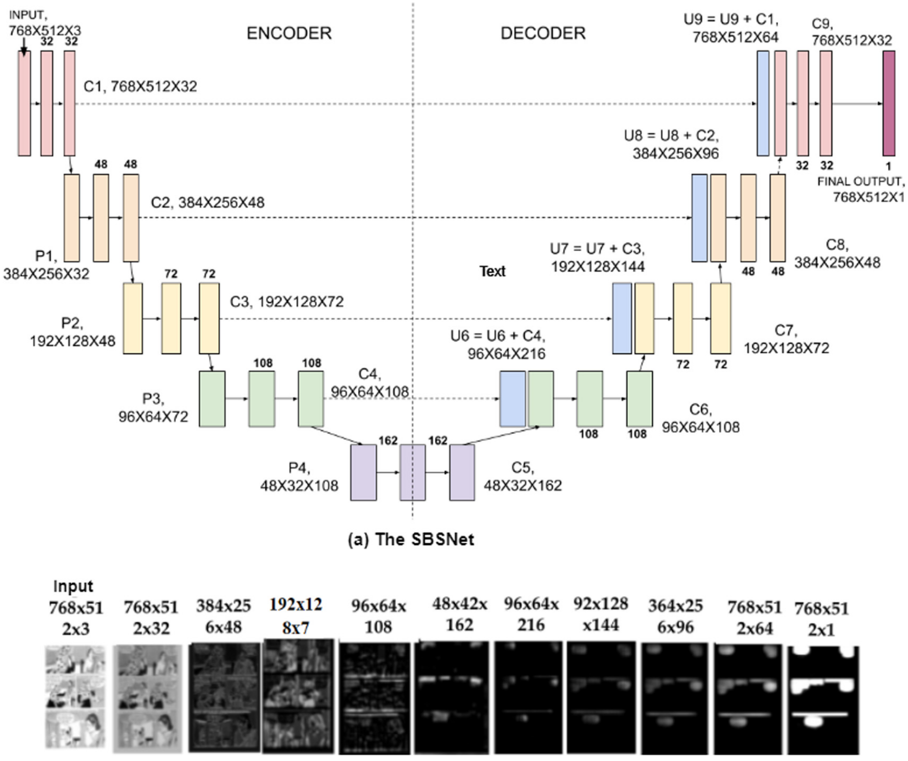
3.1 Preprocessing

A digital comic book is composed of pages containing elements like panels, speech balloons, etc. Comic PDFs are first preprocessed by extracting each page as an image with a dpi parameter of 300. IMCDB's comic panels are rectangular. However, due to the digitization of old comics, some comics do not have correctly defined borders of the panel. Therefore, before testing, we have introduced borders to the comic page and remove noise from the images.

3.2 Extraction of Comic Elements

Comic Panel Extraction (CPE): In C2VNet, the comic panel extraction (CPE) task (see Fig. 2) is achieved by CPENet (see Fig. 3), which is adapted from YOLOv3 [20]. We first divide the input image into $S \times S$ grid cells and predicts bounding boxes for each cell. Each bounding box contains five predictions (x, y, w, h, C). Here x and y are center coordinates, while w and h are the width and height of the bounding box. All these coordinates are relative to image size. The confidence score (C), for each bounding box, shows how confident a model is to have a class label in that bounding box and is defined as $C = P(O) \times B$. Here, $P(O)$ is the probability of object present in the image, and B is defined as the IOU (predicted and ground-truth bounding box).

CPENet has two darknet-53 models stacked together to form a 106 layer CNN. The first model acts as a feature extractor, and the second acts as a detector. To extract multi-scale features at multiple layers, a 1×1 kernel is applied on a feature map of different sizes for the detection. Kernel detection is of shape $(1 \times 1 \times (B \times (5 \times class)))$, where B is the bounding box, and 5 is bounding boxe's five predictions. Since we are detecting one class (panels) and three bounding boxes per grid cell, our kernel shape becomes $= (1 \times 1 \times 18)$. We discuss the panel extraction in detail in Sect. 4.1.



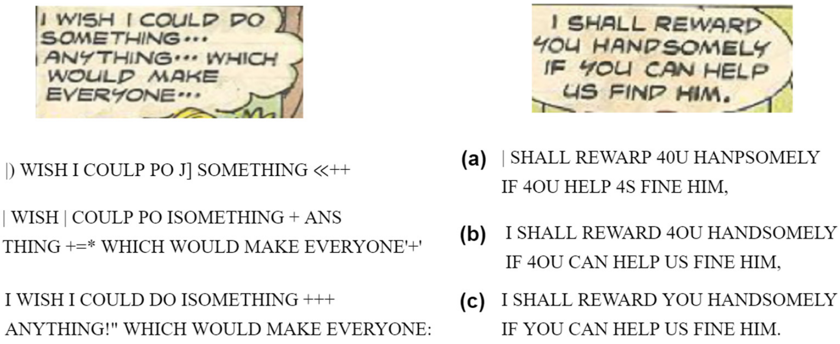
(a) The SBSNet

(b) Output of intermediate layers of the SBSNet

Fig. 4. An illustration of the proposed SBSNet and intermediate results.

Speech Balloon Segmentation (SBS): C2VNet has adapted the U-Net [25] architecture for the speech balloon segmentation (SBS) task (see Fig. 2). The architecture (SBSNet) is divided into two sub-networks (see Fig. 4a), encoding and decoding. An encoder network is used to capture the context in an image. Each block in this section applies two 3 × 3 convolutional layers followed by one 2 × 2 max-pooling down-scaling to encode the input image to extract features at multiple levels. The number of feature maps doubles in the standard UNET with a unit increase in encoding level. Therefore the architecture learns "What" information in the image, forgetting the "Where" information at the same time.

The decoder network is the second half of this architecture. We apply transposed convolution along with regular convolutions in this expansion path of the decoder network. This network aims to semantically project the discriminative feature (lower resolution) learned by the encoder onto the pixel space (higher resolution) to achieve a dense classification. The network here learns the


I WISH I COULD DO SOMETHING... ANYTHING... WHICH WOULD MAKE EVERYONE...
I SHALL REWARD YOU HANDSOMELY IF YOU CAN HELP US FIND HIM.

(a) |) WISH I COULP PO J] SOMETHING ≪++

(b) | WISH | COULP PO ISOMETHING + ANS THING +=* WHICH WOULD MAKE EVERYONE'+'

(c) I WISH I COULD DO ISOMETHING +++ ANYTHING!" WHICH WOULD MAKE EVERYONE:

(a) | SHALL REWARP 40U HANPSOMELY IF 4OU HELP 4S FINE HIM,

(b) I SHALL REWARD 4OU HANDSOMELY IF 4OU CAN HELP US FINE HIM,

(c) I SHALL REWARD YOU HANDSOMELY IF YOU CAN HELP US FINE HIM.

Fig. 5. Output of text detection by (a) Pytesseract [3], (b) Pytesseract after pre-processing, and (c) DTR.

forgotten "Where" information by gradually applying upsampling. The number of levels in the encoder network is equal to the decoder network's number of levels. Figure 4b depicts the intermediate result of the SBS task.

Text Localization and Recognition (TLR): The TLR is one of the most challenging tasks due to variations in fonts and writing styles. The comic images also become noisy and distorted due to digitization and aging. We analyze different existing models for this task. We have experimented with Pytesseract, deep and shallow models to determine the optimum framework. Neither Pytesseract (even with pre-processing like filtering, thresholding, and morphological operation) nor the shallow models yield satisfactory results. Figure 5 depicts qualitative results of the TLR task.

For TLR, we adapted a deep learning approach (henceforth referred as Deep Text Recognizer (DTR)). We applied two deep models [8,26] as part of DTR. The CRAFT model [8] localizes the text while the Convolution recurrent neural network (CRNN) [26] model recognizes it. Text Recognition based on CRNN, which process image in 3 layers. First convolutional layer process images and extract the feature sequence. The recurrent network predicts each feature sequence; the recurrent network has bidirectional LSTM (Long short-term memory), combining forwarding and backward LSTMS. The output of LSTMs goes into the transcription network, which translates the prediction of LSTM into word labels. The output shown in Fig. 5 depicts that the proposed DTR model is performing the best as compared to the other approaches.

3.3 Video Asset Creation

Video assets are various components like audio, frame timings, transitions, and subtitles, which are essential components for creating a real video.

Text to Speech Conversion: To convert recognized text from each panel to speech (refer to Fig. 2), we have incorporated the gTTs (Google Text to Speech engine) [4] into C2VNet. The .mp3 file generated by gTTs is embedded into the single panel video as an audio source. The audio duration of each mp3 file is taken as one of the input by the subtitle file module.

Frame Duration Prediction: This module's main motive is to predict the subtitle file's frame duration when given a word count. A linear regression model is the best fit to find the relationship between these two variables. The dataset used is the English subtitle file of the movie "Avengers". Data preprocessing involves removing punctuation and symbols that are not counted as words or converted to speech. Dataset consists of 2140 instances for word count and frame duration, split into a 3:2 ratio for training and testing.

$$\text{Predicted frame duration} = \theta_1(\text{word count}) + \theta_2 \tag{1}$$

where, the parameters $\theta_1 = 1.199661$ and $\theta_2 = 0.134098$. To evaluate the model (1), error is calculated as Mean Square error which comes out 0.29488. The model, combined with text to speech module predicts frame duration for IMCDB recognised.

Subtitle File: To generate a subtitle (.srt) file of a complete comic book video as shown in Fig. 2, we have considered two inputs. The first one is the text file generated by the text recognition module of each panel. The second one

Algorithm 1. Animation creation algorithm

Input: P, M, t
Output: V

```
procedure SYNTHESZIEANIMATION($P, M, t$)                      ▷ Animation synthesis
    $Contour \leftarrow$ Contour on mask                               ▷ Using M
    Draw Contour on original panel image
    $A \leftarrow Area(Contour)$
    if $A > 0.02 \times M$ then
        $V_C \leftarrow Contour$                                        ▷ Valid Contour
    $S_C \leftarrow SortContour(V_C)$   ▷ Sort contours with respect to their position in the page
    while $C \leftarrow S_C$ do                                      ▷ For each contour
        $L \leftarrow ArcLength(C)$
        $X = approxPolyDp(C, L)$
        if $Length(X) > 6$ then
            $DrawContour(C, Red)$                                     ▷ Narration Box
        else
            $DrawContour(C, Yellow)$                                  ▷ Speech Balloon
    $FPS \leftarrow 10$                                     ▷ Video synthesis parameter
    $Frame_{Loop} \leftarrow t \times 5$                            ▷ Video synthesis parameter
    $V \leftarrow CreateVideo(FFMPEG, P, C)$
    return $V$                                            ▷ The Synthesized video is V
```

is the minimum of the audio duration from text to speech module and the regression prediction model. We have iteratively updated the subtitle file corresponding to each panel. The subtitle file is consisting of textual description representing the frame number, duration and the text spoken as part of narration box/speech balloon, i.e. the voice over. For each frame these information is grouped into three consecutive lines in the .srt file. If there are n panels in comic book, then $3n$ number of lines will be there in a subtitle file of that comic book video. The details of the lines are as follows: first line mentions the frame number, second line mentions duration for subtitle and the third line mentions the text spoken. The frame numbering starts from 1 and the duration is of format "hours:minutes:seconds:milliseconds" with the milliseconds are rounded to 3 decimal places.

3.4 Video Composition

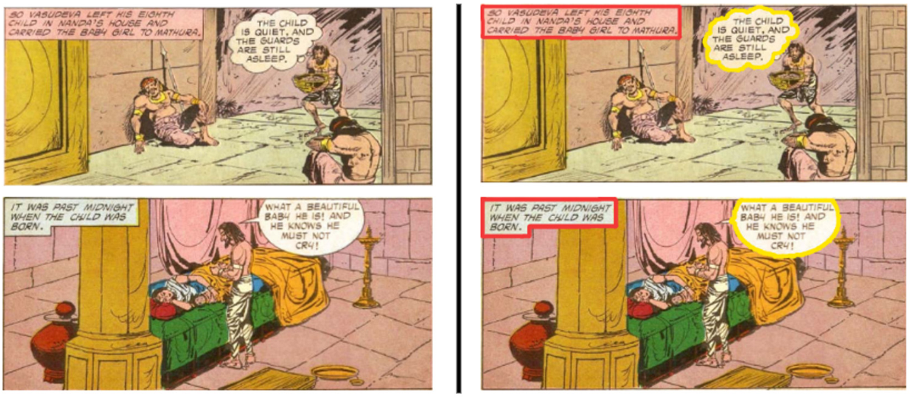
Animation: To have an animation effect for every frame of the individual panel video, we loop two images, the original panel image and another is the same panel image but with the highlighted contour of speech balloons. Algorithm 1 shows the steps of applying animation to the extracted panels. The input to the algorithm are panel image from CPE (P), speech balloon mask from SBS (M) and audio duration for panel (t), and the output is the video (V) with animation effect. Line no 8 initiates the animation process by collating all the components extracted from line 2 till this point. The thresholds for valid contour selection (line no 5) and speech balloon segmentation (line no 11) are empirically determined. Figure 6 depicts the results of the proposed animation creation algorithm on two different input panels. The images on the left (Fig. 6(a)) are the extracted panels, while the right ones (Fig. 6(b)) shows the output. Different colors are used to highlight the narration box ("Red") and the speech balloon ("Yellow").

Individual Panel Video Container: To make the video of the comic, we have utilized the "FFmpeg". It is an open-source tool used for video and audio processing. This module takes the original panel, highlighted panel image, and audio duration as input and generates a video sequence of the panel with a frame rate as per its audio duration (refer to Fig. 2). Audio of respective text files (following speech balloon ordering from left to right) for individual panels then embeds into the video. In this way, the video container of a single panel is built.

Comic Book Video: After completing all video containers of individual panels, this module, using FFmpeg, combines all the separate panel video containers to form a video with audio, subtitles, and animation (see Fig. 2).

4 Experiments and Results

The implementation prototype is developed in Python, using a standard Python library, while video processing is performed by "FFmpeg" in Python. Image processing relies on the open-source library "OpenCV" which provides standard tools for thresholding, masking, contouring, and resizing the image.


SO VASUDEVA LEFT HIS EIGHTH CHILD IN NANDA'S HOUSE AND CARRIED THE BABY GIRL TO MATHURA.
THE CHILD IS QUIET, AND THE GUARDS ARE STILL ASLEEP.
IT WAS PAST MIDNIGHT WHEN THE CHILD WAS BORN.
WHAT A BEAUTIFUL BABY HE IS! AND HE KNOWS HE MUST NOT CRY!
SO VASUDEVA LEFT HIS EIGHTH CHILD IN NANDA'S HOUSE AND CARRIED THE BABY GIRL TO MATHURA.
THE CHILD IS QUIET, AND THE GUARDS ARE STILL ASLEEP.
IT WAS PAST MIDNIGHT WHEN THE CHILD WAS BORN.
WHAT A BEAUTIFUL BABY HE IS! AND HE KNOWS HE MUST NOT CRY!

(a) Input **(b) Output**

Fig. 6. An illustration of detection and classification of speech balloon and narration boxes in two different panels.

4.1 Training and Testing

This section describes the training and testing methodologies being followed in this work for the individual tasks, i.e. CPE and SBS.

CPE Task: For the CPE task by CPENet (Fig. 3 in Sect. 3.2), first we resize all the comic pages to 416 × 416 while maintaining the aspect ratio. The first detection is 82^{nd} layer with a down-sampling image by 32. Our detection feature map here becomes $(13 \times 13 \times 18)$. Similarly, the second detection is at the 92^{nd} layer, where down-sampling is by 16, and the detection feature map is $(26 \times 26 \times 18)$. The 3rd detection is at the last 106^{th} layer with down-sampling by eight, and the feature map is $(52 \times 52 \times 18)$. For every grid cell, 3 per scale, i.e., 9 anchor boxes, are assigned. The box having maximum overlapping with ground truth is chosen. The Sigmoid classifier is used for object class prediction and confidence score. For training, we leverage transfer learning by using pre-trained weights of ImageNet on VGG16. We kept the learning rate = 0.001 and used a batch size of 64 with an IOU of 0.75. We trained using 414 comic page images of different size rectangular panels. The preprocessing stage discussed in Sect. 3.1 improves CPENet's performance.

SBS Task: We have used 1502 comic book pages and taken a training:validation: testing ratio of 70:10:20. The entire dataset is shuffled randomly to ensure that the same comic book is not used together in a single batch, resulting in overfitting the model. Resizing of pages $(768 \times 512 \times 3)$ and binary masks $(768 \times 512 \times 1)$ is also performed. Adam optimizer [15] is used with a learning rate of 0.001. The training runs for 25 epochs with a batch size of 16 and a callback function with the patience of 2. A Sigmoid activation layer is added to

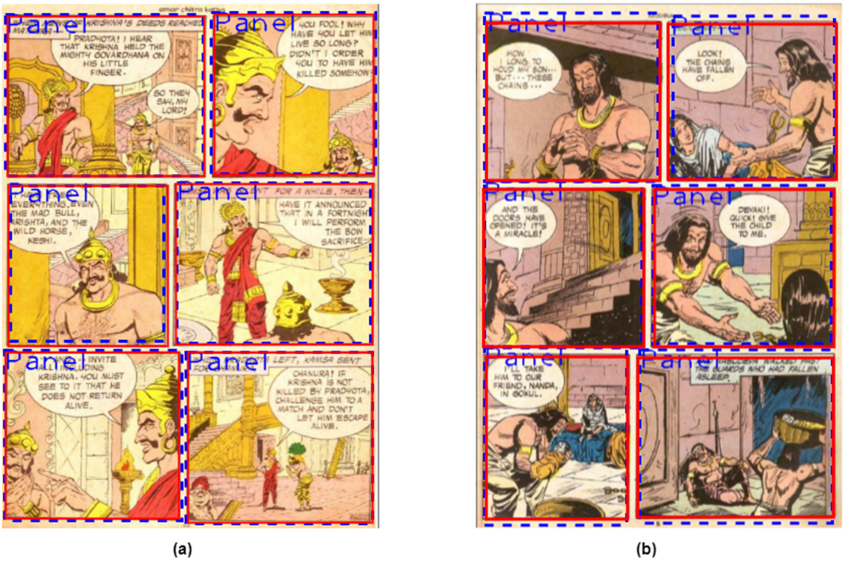

Panel
PRADYOTA! I HEAR THAT KRISHNA HELD THE MIGHTY GOVARDHANA ON HIS LITTLE FINGER.
SO THEY SAY, MY LORD!
Panel
YOU FOOL! WHY HAVE YOU LET HIM LIVE SO LONG? DIDN'T I ORDER YOU TO HAVE HIM KILLED SOMEHOW?
Panel
EVERYTHING, EVEN THE MAD BULL, ARISHTA, AND THE WILD HORSE, KESHI.
Panel
FOR A WHILE, THEN—
HAVE IT ANNOUNCED THAT IN A FORTNIGHT I WILL PERFORM THE BOW SACRIFICE:
Panel
INVITE ALL INCLUDING KRISHNA. YOU MUST SEE TO IT THAT HE DOES NOT RETURN ALIVE.
Panel
LEFT, KAMSA SENT
CHANURA! IF KRISHNA IS NOT KILLED BY PRADYOTA, CHALLENGE HIM TO A MATCH AND DON'T LET HIM ESCAPE ALIVE.
(a)
Panel
HOW I LONG TO HOLD MY SON... BUT... THESE CHAINS...
Panel
LOOK! THE CHAINS HAVE FALLEN OFF.
Panel
AND THE DOORS HAVE OPENED! IT'S A MIRACLE!
Panel
DEVAKI! QUICK! GIVE THE CHILD TO ME.
Panel
I'LL TAKE HIM TO OUR FRIEND, NANDA, IN GOKUL.
Panel
VASUDEVA WALKED PAST THE GUARDS WHO HAD FALLEN ASLEEP.
(b)

Fig. 7. Representing panel detection by model, where panel in red is the ground truth and dashed blue line is model prediction.

give the output pixel value between 0.0 and 1.0. We have empirically determined a threshold value of 0.5, greater than which a pixel is classified speech balloon (or narrative text), otherwise non-speech balloon region.

4.2 Results

We could not compare the overall framework C2VNet for audio-visual synthesis due to the unavailability of work. However, in this sub-section, we present the results and analysis of the intermediate stages.

CPE Task: Figure 7 shows correct panel detection by CPENet model. The figure represents panel detection by model, where panel in red is the groundtruth and dashed blue line is model prediction. Object detection accuracy is calculated using Precision and Recall.

$$Precision = \frac{TP}{TP + FP} \qquad (2)$$

$$Recall = \frac{TP}{TP + FN} \qquad (3)$$

where TP = True positive, FP = False Positive, FN = False negative.

Table 2. Quantitative comparison between CPENet and Nhu et al. [16].

Method	By [16]	CPENet
Accuracy	0.952	0.986
IOU score	0.944	0.977
Recall	0.990	0.997
Precision	0.952	0.978

(a) Training Performance

Method	By [16]	CPENet
Accuracy	0.932	0.979
IOU score	0.937	0.961
Recall	0.988	0.976
Precision	0.948	0.975

(b) Testing Performance

Precision and Recall are calculated using Intersection over Union (IOU), in which we define an IOU threshold and prediction with IOU > threshold are true positive and otherwise false positive. For the accuracy calculation, we took IOU threshold as 0.5. To evaluate our model's performance, we plotted a precision and recall curve that varies with each input. We used all points interpolation and calculate the area under the curve, which gives accuracy. Test dataset consists of 189 comics with ground truths containing 5 attributes ($class$, x, y, w, h) and prediction contains 6 attributes ($class$, $confidencescore$, x, y, w, h).

Comparative Analysis on CPE Task: To evaluate the model performance, we compared CPENet with the state-of-the-art model [16] published recently. We computed the results of the state-of-the-art on IMCDB by integrating available source codes. Table 2a shows the training, and Table 2b shows the testing comparison on the IMCDB dataset. The method [16] has a 1.2% better recall than our CPENet model, however, our model has 2.7% better precision and yielded a highly accurate panel extraction which is 4.5% more than [16]. This means that our proposed model CPENet can predict relevant panels over [16] as a better precision value is essential for proper video synthesis.

SBS Task: Figure 8 depicts the proposed model's qualitative results for the SBS task. It can be observed that SBSNet is able to achieve high accuracy for the given task as compared to the ground truth. Four metrics, i.e., DICE, IOU-score, Recall, and Accuracy, are followed for quantitatively evaluating the SBS task. The dice coefficient (F1 score) is two times the area overlap divided by the total number of pixels in both images. IOU-score is the ratio of logical-and and logical-or of the predicted and true output.

$$DICE = \frac{2 * TP}{TP + FP + FN} \quad (4) \qquad IOU = \frac{LogicalAnd}{logicalOr} \quad (5)$$

where TP = True positive, FP = False positive, FN = False negative.

Ablation Study: We have experimented with various filters for the network layers and reported in Table 3. The model column depicts the filters in the network

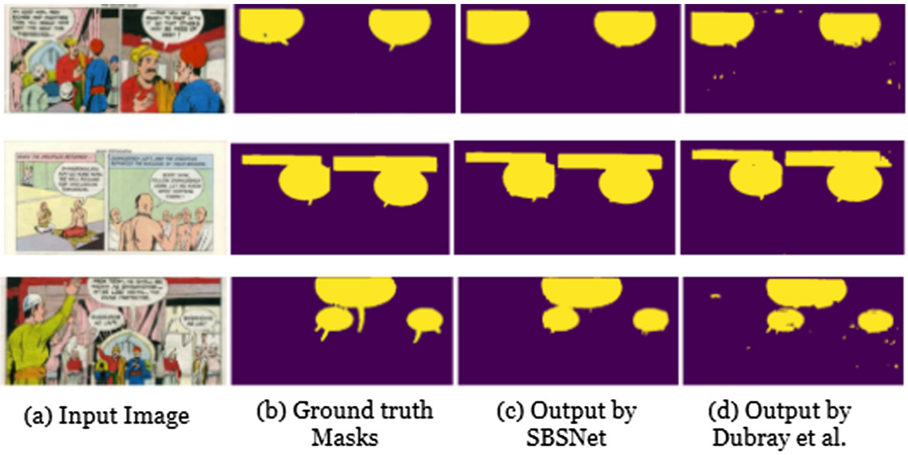


Fig. 8. Qualitative comparison between Dubray et al. [10] and SBSNet.

model. In the first model, the number of filters in the first layer is 16, which doubles in the subsequent layer to $32, 64, 128$ and 256. Total trainable parameters in this network is $1, 941, 105$. The last four columns are of the performance measures that help us to determine the best model. The performance of the model in the first and last row is approximately the same. The last one has slightly lesser accuracy, however, better in IOU-score and Recall. Still if both are considered same in the performance, the last one $582k$, i.e., $(1, 941, 105 - 1, 358, 485 = 582, 620)$ lesser trainable parameters than the first one. This means that the last model is giving us the same result with lesser resources. The second and third models are also performing well. However, the second lags in recall and the third one in IOUs-score compared to the last (4^{th}) one. Hence, we have considered the fourth model in C2VNet. The performance (accuracy, IOU score, and recall) level slightly drops (except for the precision, which increases) when we move over to testing w.r.t training and validation. We got a 98.8% accuracy along with a 0.913 IOU score, a 0.969 recall value, and a precision of 0.967 when we tested our model on IMCDB. During our model's training on IMCDB, the scores were $0.993, 0.948, 0.972$, and 0.965 for accuracy, IOU score, recall, and precision,

Table 3. Ablation study: model comparison.

Model	Trainable parameters	DICE	IOU	Recall	Accuracy
(16->32->64->128->256)	1,941,105	0.97089	0.94343	0.95633	0.99269
(8->16->32->64->128)	485,817	0.96285	0.92837	0.93744	0.99078
(16->24->36->54->81)	340,252	0.96810	0.93818	0.96274	0.99191
(32->48->72->108->162)	1,358,485	0.96927	0.94039	0.97185	0.99214

Table 4. Quantitative performance of SBSNet on IMCDB.

Result/Data	Training	Validation	Testing
Accuracy	0.993	0.989	0.988
IOU score	0.948	0.918	0.913
Recall	0.972	0.974	0.969
Precision	0.965	0.961	0.967

respectively. Table 4 presents the performance of the 4^{th} model over the training, validation, and testing set of the IMCDB.

Comparative Analysis on SBS Task. Table 5 shows the comparative study of the SBSNet with [10]. The comparison reveals that SBSNet is lighter, faster, and more accurate than [10]. The total number of parameters was reduced more than 13 times. The model given by [10] was trained and tested on IMCDB. The training time of SBSNet was reduced about 16 times, while testing time was reduced by 19.5 s. SBSNet took 13.4 s when tested on over 300 comic book pages. The size of our model is 16Mb in contrast to 216Mb by [10]. SBSNet also gives a comparable result. The accuracy, IOU score, recall, and precision are in proportion with that of [10] as seen in Table 5a.

Results of Video Synthesis: The proposed C2VNet framework generates the output as an interactive comic book video with enhanced multimedia features like audio, subtitles, and animation, where the input is given as a digitized or born digital comic book. Figure 9 depicts a few resultant video frames created from one such comic book. At the time $t = 0$ second, the first extracted panel of comic storybooks is made as the first frame of the video followed by its highlighted contour frame. Following this, all single panel video frames are combined with their respective durations forming a complete video of the comic storybook. As indicated in Fig. 9 as dummy signal waveform, the audio information, generated using the technique described in Sect. 3.3 is also embedded into the video frames. The reader has the option to toggle between, enable or disable the subtitle of the video, which is also part of the video file.

Table 5. Quantitative comparison between SBSNet and Dubray et al. [10].

Result/Model	**By** [10]	**Proposed**
Accuracy	0.984	0.988
IOU score	0.877	0.913
Recall	0.920	0.969
Precision	0.949	0.967

(a) Performance

Result/Model	**By** [10]	**Proposed**
Parameters	18,849,033	1,358,485
Training time	6781 sec	431 sec
Testing time	32.9 sec	13.4 sec
Size of output	216 Mb	16 Mb

(b) Network model

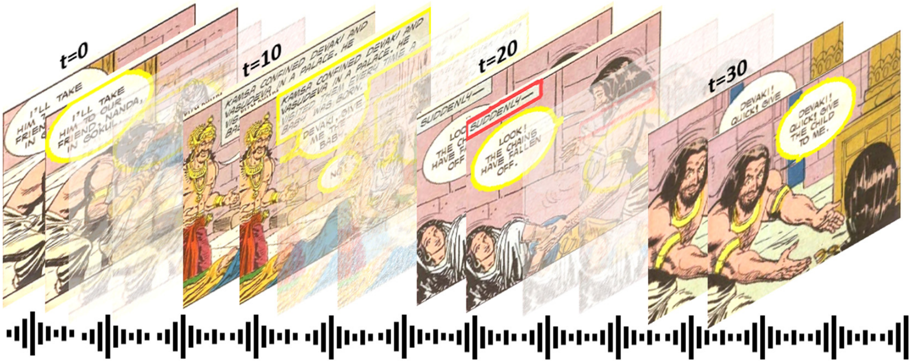

t=0
I'LL TAKE HIM TO OUR FRIEND, NANDA, IN GOKUL.
t=10
KAMSA CONFINED DEVAKI AND VASUDEVA IN A PALACE. HE VISITED THEM EVERY TIME A BABY WAS BORN.
DEVAKI, GIVE ME THE BABY.
NO!
t=20
SUDDENLY—
LOOK! THE CHAINS HAVE FALLEN OFF.
t=30
DEVAKI! QUICK! GIVE THE CHILD TO ME.

Fig. 9. An illustration of an output video generated by C2VNet.

5 Conclusion

This paper proposed a framework "*Comic-to-Video Network (C2VNet)*", which evolves panel-by-panel in a comic strip and eventually creates a full-length video of a digitized or born-digital storybook embedded with multimedia features like audio, subtitle, and animation. To support our work, we proposed a dataset named "*IMCDB: Indian Mythological Comic Dataset of digitized Indian comic storybook*" in the English language with complete annotation for panels, binary masks of the text balloon and text files for each speech balloon and narration box within a panel and will make it publicly available. Our panel extraction model "*CPENet*" shows more than 97% accuracy, and the speech balloon segmentation model "*SBSNet*" gives 98% accuracy with the reduced number of parameters, and both performed better than state-of-art models. C2VNet is a first step towards the big future of automatic multimedia creation of comic books to bring new comic reading experiences.

References

1. Digital comic museum. https://digitalcomicmuseum.com/
2. IMCDB dataset. https://github.com/gesstalt/IMCDB
3. Pytesseract. https://pypi.org/project/pytesseract/
4. Text-to-speech. https://cloud.google.com/text-to-speech
5. Aizawa, K., et al.: Building a manga dataset "manga109" with annotations for multimedia applications. IEEE Multimedia **2**, 8–18 (2020)
6. Arai, K., Tolle, H., Arai, K., Tolle, H.: Method for real time text extraction of digital manga comic, pp. 669–676 (2011)
7. Aramaki, Y., Matsui, Y., Yamasaki, T., Aizawa, K.: Text detection in manga by combining connected-component-based and region-based classifications. In: ICIP, pp. 2901–2905 (2016)
8. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: CVPR, pp. 9357–9366 (2019)

9. Bergs, A.: Protanopia, a revolutionary digital comic for iPhone and iPad (2018). http://andrebergs.com/protanopia
10. Dubray, D., Laubrock, J.: Deep CNN-based speech balloon detection and segmentation for comic books. In: ICDAR, pp. 1237–1243 (2019)
11. Guérin, C., et al.: eBDtheque: a representative database of comics. In: ICDAR, pp. 1145–1149, August 2013
12. Ho, A.K.N., Burie, J., Ogier, J.: Panel and speech balloon extraction from comic books. In: DAS, pp. 424–428 (2012)
13. Iyyer, M., et al.: The amazing mysteries of the gutter: Drawing inferences between panels in comic book narratives. In: CVPR, pp. 6478–6487 (2017)
14. Khan, F.S., Anwer, R.M., van de Weijer, J., Bagdanov, A.D., Vanrell, M., Lopez, A.M.: Color attributes for object detection. In: CVPR, pp. 3306–3313 (2012)
15. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: ICLR (2015). http://arxiv.org/abs/1412.6980
16. Nguyen Nhu, V., Rigaud, C., Burie, J.: What do we expect from comic panel extraction? In: ICDARW, pp. 44–49 (2019)
17. Ogawa, T., Otsubo, A., Narita, R., Matsui, Y., Yamasaki, T., Aizawa, K.: Object detection for comics using manga109 annotations. CoRR (2018)
18. Pang, X., Cao, Y., Lau, R.W., Chan, A.B.: A robust panel extraction method for manga. In: ACM MM, pp. 1125–1128 (2014)
19. Ponsard, C., Ramdoyal, R., Dziamski, D.: An OCR-enabled digital comic books viewer. In: ICCHP, pp. 471–478 (2012)
20. Redmon, J., Farhadi, A.: Yolov3: an incremental improvement. arXiv (2018)
21. Rigaud, C., Burie, J., Ogier, J.: Segmentation-free speech text recognition for comic books. In: ICDAR, pp. 29–34 (2017)
22. Rigaud, C., Burie, J., Ogier, J., Karatzas, D., Van De Weijer, J.: An active contour model for speech balloon detection in comics. In: ICDAR, pp. 1240–1244 (2013)
23. Rigaud, C.: Segmentation and indexation of complex objects in comic book images. ELCVIA (2014)
24. Rigaud, C., Pal, S., Burie, J.C., Ogier, J.M.: Toward speech text recognition for comic books. In: MANPU, pp. 1–6 (2016)
25. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: MICCAI, pp. 234–241 (2015)
26. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. TPAMI **39**, 2298–2304 (2017)
27. Tanaka, T., Shoji, K., Toyama, F., Miyamichi, J.: Layout analysis of tree-structured scene frames in comic images. In: IJCAI, pp. 2885–2890 (2007)
28. Tolle, H., Arai, K.: Automatic e-comic content adaptation. IJUC **1**, 1–11 (2010)
29. Yamada, M., Budiarto, R., Endo, M., Miyazaki, S.: Comic image decomposition for reading comics on cellular phones. IEICE Trans. Inf. Syst. **87**, 1370–1376 (2004)
30. Zhou, X., et al.: EAST: an efficient and accurate scene text detector. In: CVPR, pp. 2642–2651 (2017)