# Pipeline to create an English captioned video from a video with Chinese text in a given screen region

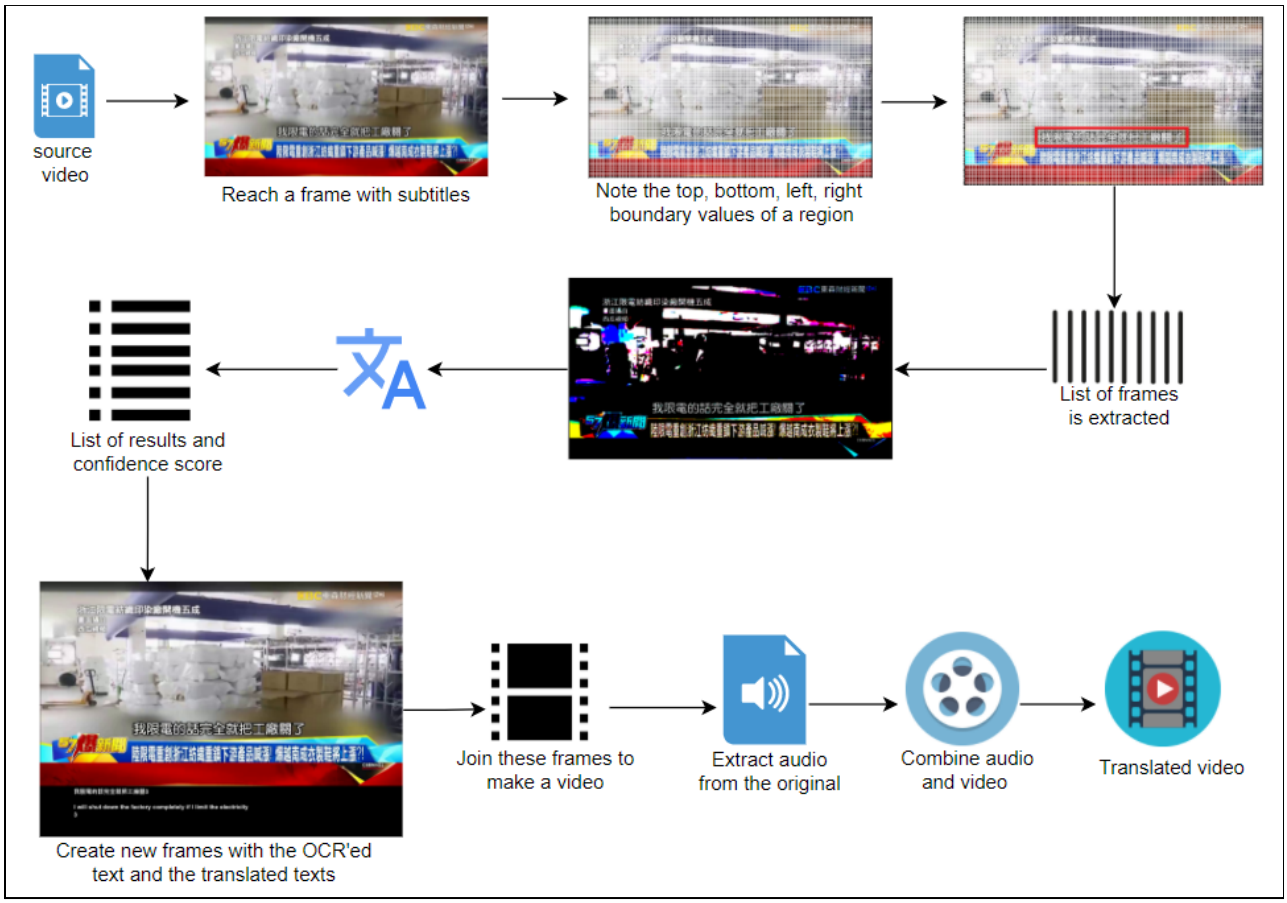
**Mehul Goel**
mg4260@columbia.edu

Directed Research Report - Fall 2021
Adviser: Professor John R. Kender, Department of Computer Science, Columbia University

Link to the files and results: drive.google.com The translated video versions have filenames starting with 'tr_'

## 1. Introduction

In order to compare Chinese news videos to that of any other country it is necessary to translate them to a common language and English makes the most sense. So this pipeline extracts and translates Chinese text from a region that the user specifies and creates a video with the translated version of the text attached below the running video. As of now, this works best with subtitles that are written in white and are not too condensed character-width wise.




Workflow of the pipeline

# 2. Pipeline User Manual

## 2.1 Get a video

Have the video on your computer or download one from youtube.

## 2.2 Selecting the region to pick text from

Keep re-running the cell to see subsequent `next_frame` to see visually what should be the pixel values for the four boundaries of the region from where the text has to be picked

```
cap = cv2.VideoCapture(video_name)
# next_frame_count = 1
starting_frame_number, starting_frame = next_frame(cap)
```

```
iteration = 1
next_frame_count=0

def next_frame(cap):

  global next_frame_count
  global iteration

  next_frame_count = 0

  # count = 0
  image=None
  while next_frame_count != 15*iteration :
    success,image = cap.read()
    next_frame_count=next_frame_count+1
  starting_image=image
  cv2_imshow(starting_image)

  iteration = iteration + 1

  return next_frame_count - 1, starting_image
```


A figure like this above will appear for you to read the top, bottom, left and right boundaries using the scales on x and y axes

### 2.3 Create a list of frames

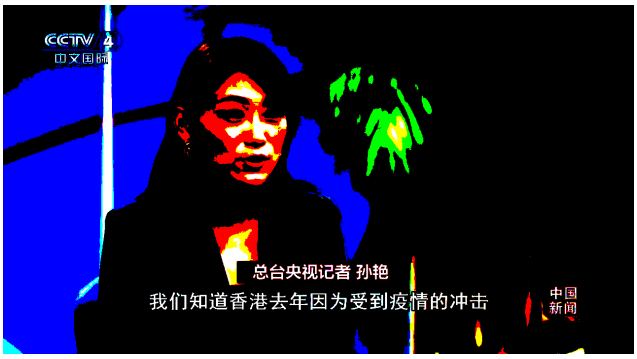
Run the cell that creates a list of frames where every 15th frame is stored in a numpy array form. Done using opencv

```
list_of_frames = get_every_15th_frame(full_video_path)
```

### 2.4 Keeping frames with distinct subtitles

Run the corresponding cell to sift through the list of frames from the video and select only the frames having distinct subtitles so that time can be saved from translating the same piece of text repeatedly. This is done by first cropping the rectangular region specified. Then that cropped image is binarized[1] to make the subtitle text stand out from the background, making it easier for the OCR to run on it which will happen later on.


Original image




Binarized image

Then the binarized image is compared with that of the previous frame and if they are similar then this new frame is ignored. Otherwise this frame is scanned for text by easyocr[2] looking for Chinese and English text and the results are stored in a format as shown below.

```
[129, '我們估計今年2021年應該是有年度的增長', 0.5912623560443827]
```

It is the frame number of the video divided by 15 as every 15th frame was picked earlier, followed by the Chinese text and lastly the confidence score.

- Comparison was done using the module `sewar`-which has many ways to compare two images, I chose the rmse way.
- Text extraction was done using the module `easyocr`-which has multilingual support and in that I had chosen Chinese and English text to be extracted
- Because of binarization being used currently only white subtitles' text can be extracted but an easy improvement would be to extract text from the original image and only if by a heuristic it is detected that the text recognized is not accurate or complete then do another extraction after binarization

### 2.5 Translating the text

Using google translator module deep-translator[4] all the Chinese texts are translated. The corresponding cells clean the text meaning they skip over the numbers, remove the special characters and spaces as deep-translator allows only alpha-type characters in the text sent to it for translation.

After translation the list of results looks like a list of the following where the english bit at the end is the new addition

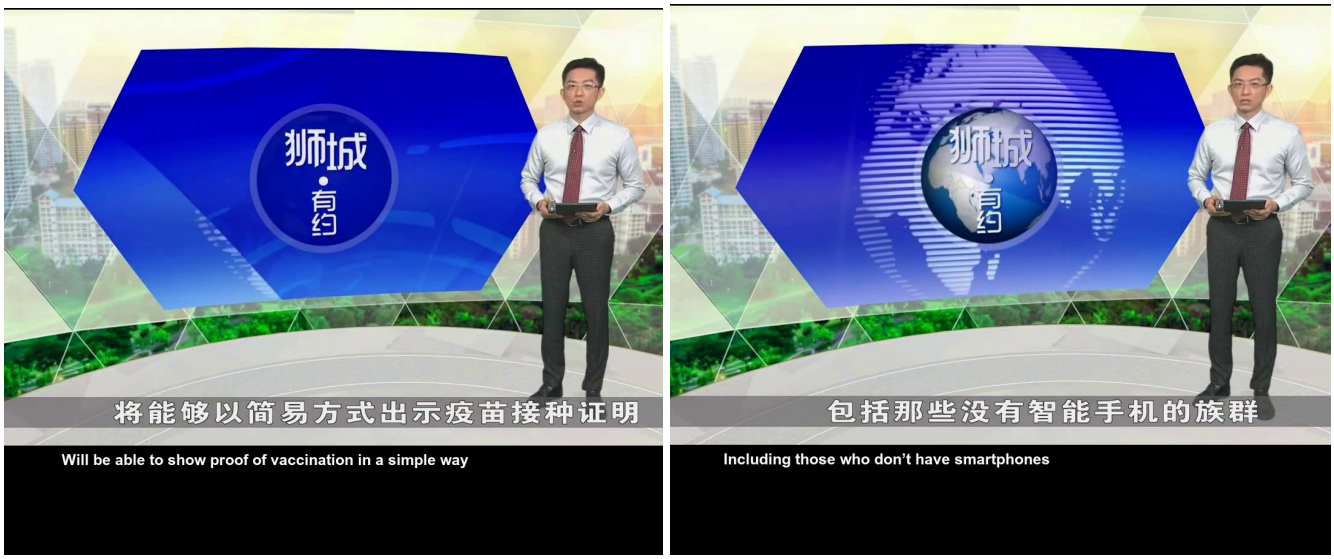
```
[129, '我們估計今年2021年應該是有年度的增長', 0.5912623560443827, ['We estimate this year', '2021', 'Year should have annual growth']]
```

### 2.6 Refining the translated text

Often slightly varying translations get created from the same subtitle in the video, this is refined by picking the one with the higher confidence score. This can be expanded to be able to replace each occurrence of a line of text, consecutive or not, with the one with the highest confidence score. Similarity between two pieces of text is checked using the `difflib's SequenceMatcher`[5]

### 2.7 Creating bordered frames

Next for each frame in the video some space is added below the video to put the Chinese text extracted and the English text in and the aforementioned text is pasted in that space. Done using the python's imaging library `Pillow`[6]. The font required also should be present in the current working directory and it should be a unicode font to be able to work with the chinese characters. I am using `Arial-Unicode-Bold.ttf` which I a downloaded from the net and place it in the working directory




Frames with extra space and translated text added

### 2.8 Creating the video

Next, the bordered frames are put together to make a video. Audio is extracted from the original video and put with the video, creating the final video. Done using the module `moviepy`[7]

## 3. Improvements

### 3.1 Selecting the starting frame

In a news video there are often animations at the start of the video, so I tried to detect when those animations have subsided and plain news narration has begun. I did it by setting a condition that when text kept appearing in a

certain location consistently the start of the time whence that started happening can be identified as the starting frame. But there were issues like at multiple places on the screen there'll be such kinds of texts. If one then restricts the location to search this text then too there can be texts like the channel name or weather or the time showing texts. So there's still scope for this feature to be added.

### 3.2 Detecting and correcting for condensed text

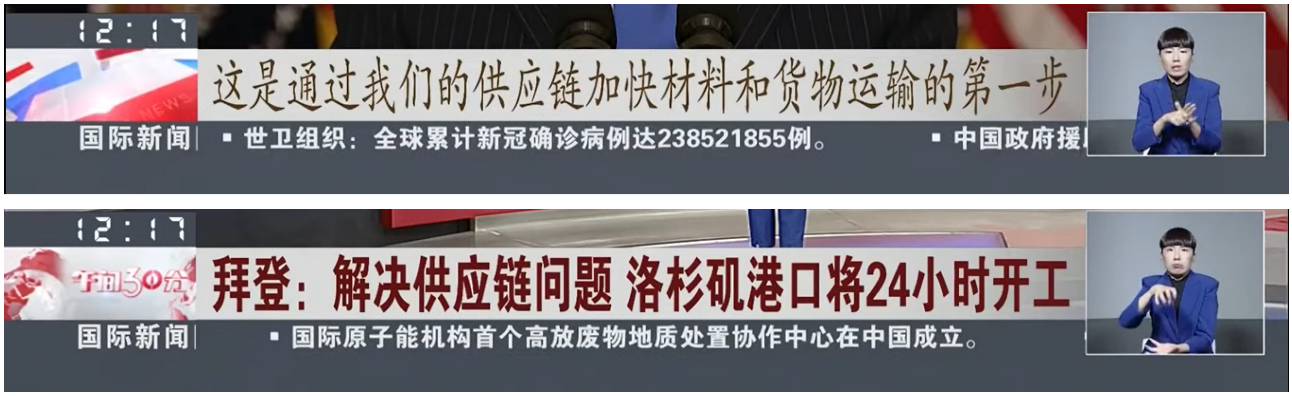

Often in news channels the headline will be shown in a condensed font and the easyocr[1] does not recognize this accurately. So an approach to solve this could be to detect the width of any one character in this condensed text and resize this image so that the character width and height matches that of a standard subtitle. I could achieve the latter but it was a hit and miss while trying to detect the width of any one character in the condensed text form. So this can be improved.

### 3.2 A way to capture the following in separate channels even though collected at once by the OCR's result, while also ignoring the advertisement and channel logo and other insignificant persistence texts:

1. headlines relevant to what is on screen,
2. subtitles,
3. headlines not relevant to what is on screen
4. text in the images or videos of the news that is being presented

This was a particularly tough challenge as in every other news channel's video the placement of these lines slightly vary. Though I tried to predict by the heuristic that if there are 3 lines of text in the lower half of the screen then I can attribute the lower two to be the subtitle and the headline but then sometimes there is the news ticker present too as the third line. So the only way to make the distinction between which is which seems to be to use the amount of time a certain text is present on the screen and the way that text appears and leaves the screen (especially to distinguish a news ticker)

**3.2 Capturing news ticker/slider/crawl headlines and finding a way to accurately find the starting and ending of a sentence**




The lowermost line is being referred to here where the news crawl runs horizontally and as the style of the space between two sentences is different in different news channels too this turned out to be a tough challenge to efficiently capture a sentence's text and ignoring its presence in the subsequent frames by recognizing using only partial text that that sentence has already been considered

### 3.2 Transcribing from audio is one option but expensive

There is no open source tool that outputs a transcript from audio in Chinese language. There are services that provide multilingual transcription from an audio snippet, like sonix.ai and it is quite accurate, just that it is not free. For example, sonix.ai costs around $10 for 1 hour of transcription.

Following is the result from using sonix.ai on this video https://www.youtube.com/watch?v=fhIiq1_vrPQ

Speaker: 如果两岸开战的话第一波死伤人数恐怕会超过24万人因此呢国防部医学局明年将斥资四十一亿元提升战时的治疗效率。你觉得呢来去听听看民众的意见它是骗人的。朋友来找名目乱花钱吧又来乱掌名目乱花钱吧把我们的税金当什么省起来真开战马上就投降了要赤字这么多钱我是没有没有很赞成是感觉不需要了没有这个必要就是。就是开战地王投降了不需要花这个钱神气起来真开战马上就投降了。这个因为我觉得我们这马上就会投降了所以就不用花这个钱就他和他2.4万死伤了太惨你的希望就是不要这样不要发生这种事我反正是谁把台湾推向战争这就是现在的那个民进党政府啊无言对这个政府真的很无言。当一个政府的领导人应该是想办法尽量怎么样去避免两岸发生冲突而不是用这种话来欺骗人民。吗。当然会呀然后小孩这么想那你会担心两岸开战吗。不会我想吧因为不会开战美国会保护我们绝对不可能啦美国美国绝对不可能来帮真正的往台湾我所以从24万人他们都爱我没有我没有想过这件事情好可怕老天啊就是以中共的武力然后大国政治来讲话美国应该是不会美日应该是不会真的帮台湾的都只是嘴巴上讲讲而已我不会上网因为不关我的事这样我就不会啊因为我觉得不会打到台湾了对我是觉得不会打台湾因为有很多国家可能更需要被打吧这会太直接应该现在年轻人应该很讨厌战争不是吗。没有人喜欢打仗吧就这么简单。冬季奥运就知道了如果台湾民众再不觉醒那么两岸之间真的会进入另外一个状况型态了真的不要再玩意识形态没有意义。那我会好啊。我非常不自大完全对啊他啊是啊就好现在啊想办法了我节日里啊所以我听说过可能真的不幸打起来的话可能几个小时之内就结束了这样没等着吧。你不要以为噢一时之快我们人民苦啊对我的那一端站起来他会不会跟着我们车站搞不好逃掉了仙桃的就是他了。我觉得做足做一个立得一定要立得一个上司审起来真开战马上就投降了事最多民众选的答案好针对这41亿你觉得该不该花呢。留言告诉我们。

If there is a war between the two sides of the strait, the number of casualties in the first wave may exceed 240,000. Therefore, the Medical Bureau of the Ministry of National Defense will spend 4.1 billion yuan next year to improve the efficiency of wartime treatment. Do you think it is deceptive to come and listen to the opinions of the people? Friends come to find names and spend money indiscriminately that is. The king surrendered when the battle started, and he didn't need to spend the money. This is because I think we will surrender right away, so we don’t need to spend this money. He and 24,000 were killed and injured. It’s too miserable. Your hope is not to do this, not to happen. I am the one who pushed Taiwan to war. This is now. The DPP government of ah, speechless, is really speechless to this government. When the leader of a government tries to avoid conflicts between the two sides of the strait as much as possible, instead of deceiving the people with such words. . Of course it will. Then the child thinks so. Are you worried about a war between the two sides of the strait? No, I don’t think it’s because it won’t go to war. The United States will protect us. It’s absolutely impossible. The United States is absolutely unable to help me go to Taiwan. So from 240,000 people, they all love me. I haven’t thought about it. It’s terrible. Ah, it's just by the force of the Chinese Communist Party after great power politics, the United States should not be able to speak. The United States and Japan should not really help Taiwan. It’s just talking about it. I don’t know how to go online because it’s not my business, so I won’t because I don’t think I can fight. When I arrived in Taiwan, I felt that I would not fight Taiwan because there are many countries that may need to be beaten even more. It would be too straightforward. Now young people should hate war, right? No one likes to fight. It's that simple. The Winter Olympics will know that if the people of Taiwan do not awaken, then the two sides of the strait will really enter another state of affairs. It really makes no sense to stop playing with ideology. Then I will be fine. I'm very arrogant and totally right, he, yeah, just think of a way now. I’m on holiday. So I’ve heard that if it is really unfortunate, the fight may end within a few hours, so I didn’t wait. Don't you think oh, all of a sudden, our people are suffering and standing up on my end, he will follow us at the station, maybe he is the one who escaped from Xiantao. I think it is necessary to do a good job and to stand up so that a boss must surrender immediately when the trial starts. The most popular answer is for the 4.1 billion you think you should spend it. Leave a message to tell us.

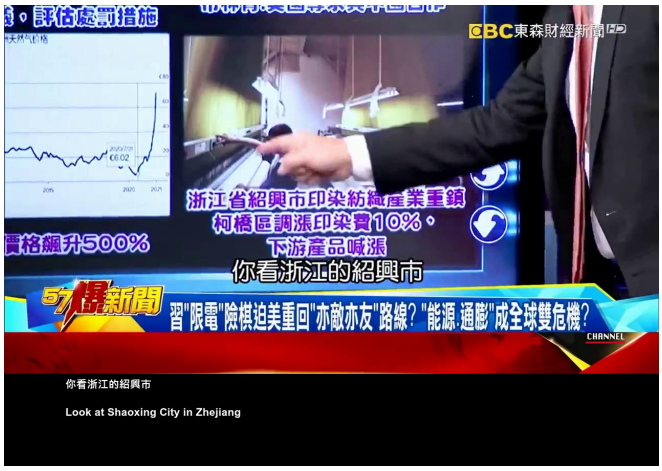
## 4. Results

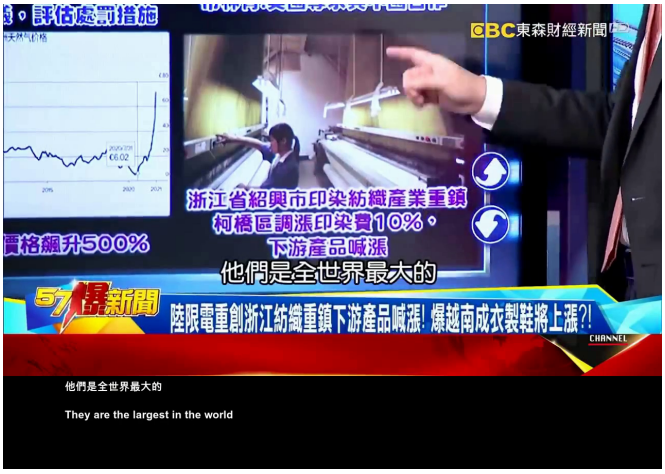
Please check out the videos on this link for the results:
https://drive.google.com/drive/folders/1ihoRm7JmH3m0LTeKCUVni7PlP_GGlEod?usp=sharing

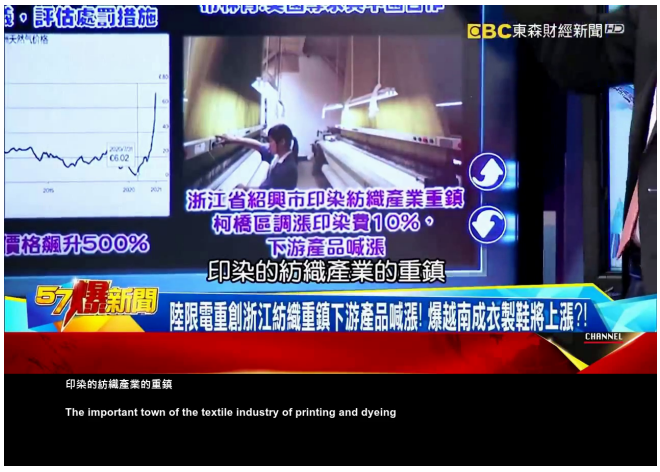
The translated video versions have filenames starting with ‘tr_’

Screen captures from a small clip are shown below:
It tells the following story:

> Look at Shaoxing City in Zhejiang. They are the largest in the world. The important town of the textile industry of printing and dyeing. That especially this place in Keqiao District
> He said he didn't expect Zhejiang province to ask me to cut off the electricity.  It will shut down the factory completely if they limit the electricity. So look at the current operating rate. Probably about 4 can't make it 5 into the left and the right . He said that I want to increase the printing and dyeing fees by 10%, this creates a butterfly effect. The downstream must also rise.





他就說沒想到浙江省

He said he didn't expect Zhejiang Province


就要求我限電

Asked me to cut off the electricity


我限電的話完全就把工廠關3

I will shut down the factory completely if I limit the electricity
3


所以你看他們現在的開機率

So look at their current operating rate


大概約莫是4成不到5成左右

Probably about 4 Can't make it 5 Into the left and right


那4成不到5成左右

That 4 Can't make it 5 Into the left and right





## 5. References


1. Image Thresholding or binarization - https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html
2. easyocr - https://github.com/JaidedAI/EasyOCR
3. sewar - https://pypi.org/project/sewar/
4. deep-translator - https://github.com/nidhaloff/deep-translator
5. SequenceMatcher - https://docs.python.org/3/library/difflib.html
6. Pillow - https://pillow.readthedocs.io/en/stable/