Usage Documentation

RTMPRetroTV

August 5, 2024

Abstract

This document provides detailed usage instructions and examples for the *RTMPRetroTV* tool. It includes installation instructions and additional notes to help you get started.

Contents

1 Introduction

RTMPRetroTV is a command-line tool designed to simplify the process of creating your own old-school broadcast channel. It leverages `ffmpeg` to stream video content to an RTMP server, allowing users to emulate the experience of traditional TV broadcasts.

1.1 How It Works

- **YAML Configuration:** Users define a schedule in a YAML file, listing video files and the exact times they should be streamed. This configuration acts as the programming guide for your channel.

- **Automated Streaming:** RTMPRetroTV reads the YAML file and uses `ffmpeg` to automatically stream each video to the designated RTMP server at the specified times. This ensures that your content plays like a traditional TV channel, complete with scheduled programming.

- **Use Cases:** Ideal for anyone looking to recreate the nostalgic experience of scheduled TV programming, RTMPRetroTV is perfect for themed broadcasts, retro TV channels, or continuous playlists that mimic the feel of classic television.

By simplifying the process of setting up a streaming schedule, RTMPRetroTV enables users to deliver content in a way that harkens back to the golden age of television broadcasting.

2 Installation

Describe the steps required to install your project. Include any dependencies and how to resolve them.

```
$ git clone https://github.com/07d8b5/RTMPRetroTV.git
$ cd RTMPRetroTV
$ pip install -r requirements.txt
```

3 Usage

This section explains how to use RTMPRetroTV with the provided `config/schedule.yaml` configuration file. Follow these steps to customize and run your own broadcast channel.

3.1 Prepare Your Video Files

Before configuring RTMPRetroTV, ensure that all your video files are ready:

- **Ensure Uniform Video Format:** All videos, including scheduled programs and commercials, must be encoded in the exact same format. This includes the same codec, resolution, frame rate, and audio settings. This uniformity ensures seamless playback without interruptions or compatibility issues.

- **Create a Fallback Video:** Prepare a 1-second fallback video file. This file acts as a placeholder in case of any scheduling or playback errors. Place this file in the 'videos/' directory as specified in the YAML configuration.

3.2 Review the Provided `schedule.yaml` File

RTMPRetroTV includes an example `schedule.yaml` file that outlines a setup for a TV station. Below is the structure of the file:

```
tv_station:
  name: ExampleTV
  block_size: 30
  schedule:
    - day: 0
      time_slots:
        - time: "7:08"
          program: "LooneyTunes"
          selection_method: "shuffle"
    - day: 1
      time_slots:
        - time: "7:08"
          program: "LooneyTunes"
          selection_method: "shuffle"
    - day: 6
      time_slots:
        - time: "9:14"
          program: "LooneyTunes"
          selection_method: "sequential"
        - time: "20:30"
          program: "LooneyTunes"
          selection_method: "shuffle"

paths:
  show_metadata:
    source_directory: "./"
  commercials:
    source_directory: "video/cm/"
    episodes:
      - "ILoveTokyo.mp4"
      - "Isis.mp4"
      - "Kowa.mp4"
      - "Penck.mp4"
  LooneyTunes:
    source_directory: "video/shows/LooneyTunes/"
    episodes:
      - "ScrapHappyDaffy.mp4"
      - "TokioJokio.mp4"
```

3.3 Configuration Details

- **tv_station:**
  - **name:** Specifies the name of the TV station.
  - **block_size:** CURRENTLY UNIMPLEMENTED Defines the duration (in minutes) for each programming block.

  - **schedule:** Lists the programming schedule, specifying the day of the week (0 for Sunday, 6 for Saturday) and the time slots for each program.
  - **time_slots:** Indicates the time, program name, and selection method (e.g., `shuffle` or `sequential`) for the episodes.

- **paths:**
  - **show_metadata:** Directory containing metadata files for the shows.
  - **commercials:** Path to the directory and list of commercial episodes available for insertion during breaks.
  - **LooneyTunes:** Path to the directory containing episodes for the `LooneyTunes` program.

3.4 Run RTMPRetroTV

Once you have customized the YAML configuration, run RTMPRetroTV with the following command:

```
$ cd /path/to/RTMPRetroTV
$ make run
```

This command initiates the streaming process, using the settings from your customized `config/schedule.yaml` to broadcast according to the specified schedule.

3.5 Additional Tips

- Verify that all file paths and filenames in your YAML configuration are accurate.
- Test your setup with a short schedule to ensure everything functions correctly before a full broadcast.
- Confirm that your RTMP server is operational and ready to receive streams from RTMPRetroTV.

By following these steps, you can set up RTMPRetroTV to emulate an old-school TV broadcast channel, complete with scheduled programming and commercial breaks.