

地域連携を目的としたライブ動画ストリーミングシステムの 低速回線における再生性能の改善

東畑 和真†
Kazuma Higashihata

山内 雪路†
Yukiji Yamauchi

1. はじめに

大阪工業大学は奈良県吉野郡川上村と「連携・協力に関する協定」を締結しており、現在までに本大学の学生が様々な地域の振興活動に取り組んでいる。奈良県川上村には豊富な観光資源があり、あきつの小野公園は春に様々な種類の桜が咲き乱れる観光名所である。しかし、桜の開花時期は限られており、交通アクセスが悪いため、気軽に見に行くことができない。我々の研究室では、あきつの小野公園にネットワークカメラと気象センサーを設置し、桜を映した動画とあきつの小野公園の気象情報を配信する桜ライブキャストシステム¹⁾を提供してきた。桜ライブキャストの代表的画面を図1に示す。

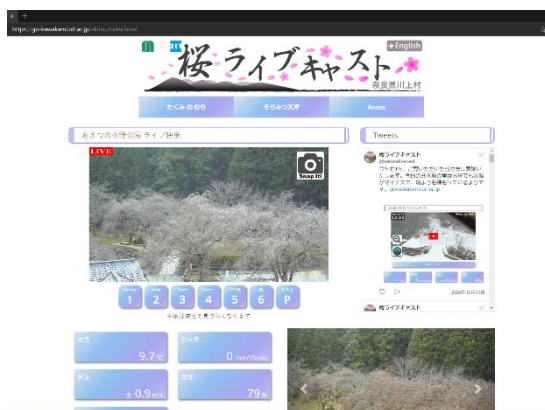


図1 桜ライブキャストの画面

桜ライブキャストの動画配信システムは、あきつの小野公園に設置されたカメラから MotionJPEG 方式で画像を一度 Web サーバーに保存する。そのカメラ画像をブラウザが一定間隔ごとに取得し、ブラウザ画面上の静止画像を差し替えることで動画再生を行っている。桜ライブキャスト機材構成図を図2に示す。

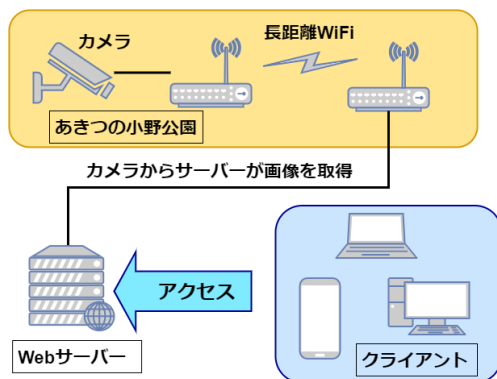


図2 ライブキャスト機材構成図

この動画配信方式を採用した理由は、カメラが PTZ 操作可能であり、映像伝送の遅延が 100 ミリ秒を超えると視覚と操作のずれを大きく感じる²⁾ことを避けるためである。

本システムは多様なユーザーに利用されており、通信速度制限された携帯回線等の低速回線を用いて桜ライブキャストに接続するクライアントが増えてきた。低速回線にて桜ライブキャストの動画再生を行った際、2つの問題が発生した。1 番目の問題は、動画再生時に画像の表示順序がまれに入れ替わることである。ブラウザが画像取得中に TCP パケットの再送が発生した場合、画像の到着順が入れ替わってしまう。2 番目の問題は、接続している回線の転送可能速度以上の画像ダウンロードによって HTTP のタイムアウトが頻発し、フレーム落ちが発生することである。

本稿では、低速回線を用いて桜ライブキャストに接続し動画再生を行った際、発生する2つの問題の改善方法の提案を行う

2. 画像の表示順補正

2.1 画像表示順の乱れが発生する問題

桜ライブキャストの動画再生は、ユーザーのブラウザ上で動く JavaScript プログラムにより一定間隔ごとにカメラの JPEG 画像を取得し、ブラウザ画面上の JPEG 画像を差し替えて実現している。

この JavaScript プログラムは1枚ごとの JPEG 画像を異なる HTTP セッションで、かつ JavaScript の非同期通信で取得しているため、画像の到着順序がまれに入れ替わることがある。ブラウザが画像取得中に TCP パケットの再送が発生した場合、再送が発生した画像より生成時刻が後の画像をクライアントが先に取得してしまうケースがあるためである。画像取得終了直後に画像を差し替えるシステムのため画像の表示時系列が順序逆転し、不自然な動画となる。パケットロスによって画像取得終了順が入れ替わる状態を図2に示す。

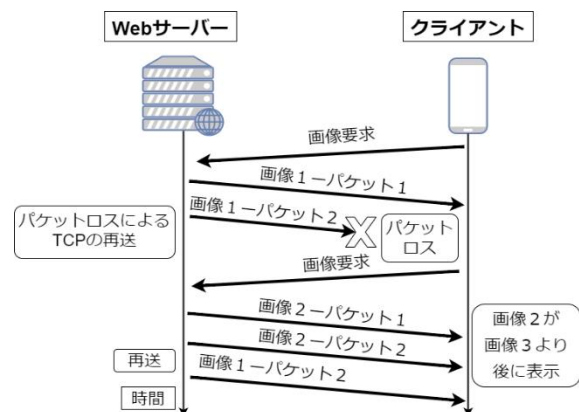


図2 パケットの到着が遅延した通信

2.2 画像表示順の乱れの改善

クライアントは取得した画像1枚ごとにサーバー側での生成時刻を参照し、画面上に表示中の画像より生成時刻の古い画像を新たに取得した場合、画面に表示しないで破棄する方法を提案する。

JPEG 画像では、「Exif³⁾」にメタデータを書き込むことができる。Exif には記録されている「DateTimeOriginal」と呼ばれる項目があり、これを参照することで生成時刻を調べられる。しかし、あきつ小野公園に設置しているカメラの画像には Exif が存在しないため、新たに Exif 情報を書き込む必要がある。Exif の書き込みは Web サーバーにてプログラムを用いて行う。カメラからサーバーへの画像転送は単一の HTTP セッションを利用したストリームで行っているため、画像の順序が入れ替わることはない。

JPEG 画像に Exif を書き込むプログラムは Python 言語を使用し、JPEG 画像に Exif を書き込む処理は外部ライブラリの「piexif」と「Pillow」を使用した。

2.3 改善結果

再送遅延による画像の取得順の入れ替わり時のシステム動作を確認する。検証環境として、大学内にカメラと「桜ライブキャスト」と同等の機能と Exif を書き込むプログラムを搭載した Web サーバーを用意した。桜ライブキャスト低速回線を再現するため、通信速度制限のかかった携帯回線のスマホでテザリングにクライアントを接続して行った。検証環境の構成図を図3に示す。

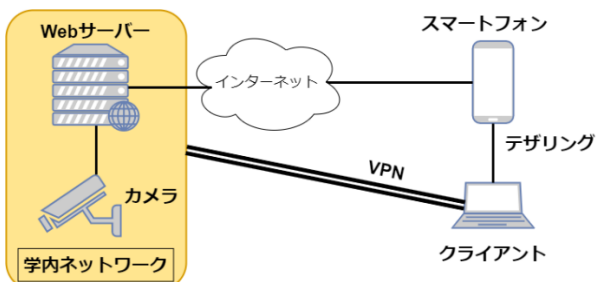


図3 検証環境の構成図

表示中の画像より古い画像の表示を阻止した際に、「更新しませんでした」と表示している画像と取得した画像の生成時刻を開発者ツールの JavaScript console 表示するようにした。その結果、画像の表示順訂正がなされていることを確認した。プログラムの動作結果を図4に示す。

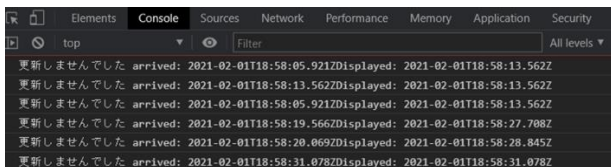


図4 画像表示順の乱れの阻止

3. 画像取得間隔の自動調整

3.1 過剰な画像要求による輻輳問題

桜ライブキャストの動画再生時、ユーザーのブラウザ上で動く JavaScript プログラムは一定間隔ごとにカメラの JPEG 画像を取得する。

しかし、画像取得は接続している回線の状況によらず同じ間隔で画像を取得するため、画像の転送に必要な帯域より接続している回線の転送帯域が下回る場合がある。その場合、動画再生においてフレーム落ちが著しく発生する。原因としてはクライアントの TCP 処理が続けられるが HTTP のタイムアウトを起こす。画像の過剰要求による HTTP タイムアウトの発生を図5で示す。

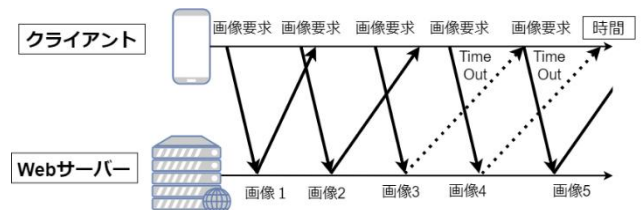


図5 過剰な画像要求による HTTP タイムアウトの発生

さらに、タイムアウトしたにもかかわらず転送されてくるパケットによって帯域が使われるため、新規に取得を試みた画像もタイムアウトになってしまう。

3.2 画像要求間隔の改善

桜ライブキャストではクライアントが1枚ずつ JPEG 画像を取得しているため、フレーム間圧縮を行なえないことから画像の転送量削減は期待できない。

そこで、画像の取得間隔を調整して転送量の削減を行う。本研究で扱うカメラは動きが少ない被写体を映しており、フレームレートを間引いても実用上問題がない動画が得られる。

本研究では、画像取得リクエストの発行から取得終了までの時間から、クライアントの通信帯域を推測し、推測した帯域を元に画像取得間隔を変更する方法を提案する。画像取得リクエストの発行から取得終了までの時間から、クライアントの通信帯域を推測し、推測した帯域を元に画像取得間隔を変更する方法を提案する。

ブラウザ上の Resource Timing API を JavaScript の「performance.getEntriesByName()」関数を用いて利用することでコンテンツのリクエストからデータの受信終了までにかかる時間を測定できる。今回は、10回分の画像取得時間の平均値で画像取得が逐次実行した際に1秒間の実行可能な画像取得リクエスト数を求め、その値を画像取得間隔の決定に利用した。今回は、10回分の画像取得時間の平均値で画像取得が逐次実行した際に1秒間の実行可能な画像取得リクエスト数を求め、その値を画像取得間隔の決定に利用した。

3.3 改善結果

次に HTTP タイムアウトによるフレーム落ち軽減の効果を確認する。検証環境は図3と同一である。クライアントが画像取得し、画像差し替え直後に取得画像の生成時刻を Exif から読み取り、ブラウザの JavaScript コンソールに表示

する。画像取得間隔の自動調整を行う場合と行わなかった場合で比較する。

画像取得間隔の自動調整を行わなかった場合の JavaScript コンソール画面が図 6、行った場合が図 7 である。画像取得調整なしでは 1 秒に 6 回の画像取得を行っているにもかかわらず最大 8 秒の間隔があいているケースが認められ、目視でもフレーム落ちを体感できる。取得間隔を自動調整した場合、1 秒間に 1 回の取得に自動的に調整され、取得した画像の生成時刻も約 1 秒間隔で取得できていることがわかる。このことから、画像取得間隔の自動調整によって、フレーム落ちの改善を行うことができた。

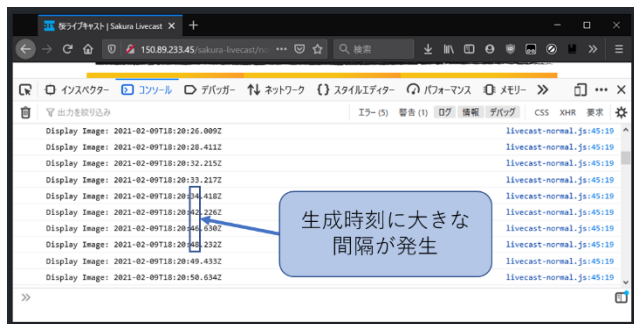


図 6 取得間隔調整なし



図 7 取得間隔調整あり

4. 結言

本稿では、奈良県川上村の観光資源の PR を目的としたライブストリーミングシステムの低速回線での桜ライブキャストの動画再生時に発生する 2 つの問題の対策法を提案した。

1 番目の問題は TCP の再送により画像の表示順が入れ替わる問題である。Web サーバーでカメラから送られた JPEG 画像に Exif を書き込み、クライアントがその Exif を読み込む。クライアントは読み込んだ Exif によって画像の時間軸を作り、表示順の補正を行った。

2 番目の問題として、画像取得に時間がかかり HTTP タイムアウトとなり、フレーム落ちが発生する。画像取得にかかる時間を計測し、その値を元に画像の取得間隔を変更するシステムを開発した。結果として HTTP タイムアウトの減少が確認できた。

低速回線での再生にて発生する 2 つの問題に対する改善案は、どちらも実用上十分な改善結果を確認できた。

参考文献

[1] 桜ライブキャスト,
[<https://go-kawakami.oit.ac.jp/akitsu/index.html?ja>], (参照 2021 年 7 月 19 日)

[2] テレイグジスタンスの研究(第 63 報)-TELESAR3 において許容される通信遅延の検討-,
[https://tachilab.org/content/files/publication/study_on_telexistence/te063.pdf], (参照 2021 年 7 月 10 日)

[3] デジタルスチルカメラ用画像ファイルフォーマット規格 Exif 2.3,
[http://www.cipa.jp/std/documents/j/DC-008-2010_J.pdf], (参照 2021 年 7 月 11 日)