1Q-06

高効率ゲーム動画配信のための ユーザ関心領域を考慮した転送データ削減手法

駒牧 潤也† 中島 克人‡
東京電機大学大学院 未来科学研究科 情報メディア学専攻†‡

1 はじめに

インターネット上での動画コンテンツは配信プラットフォームの普及や巣籠り需要の影響から活用される機会が増加している．特に，ゲームプレイを中心としたライブ配信を提供する Twitch では 2020 年から視聴者が急増している[1]．動画を視聴者側のデバイスに送信する際の問題として，データ容量の大きさや視聴者数増加に伴うトラフィックの増大が挙げられる．動画配信サーバの処理能力や視聴者側のネットワーク環境がボトルネックとなる場合，輻輳が発生して動画の再生が停止したり，低い解像度で送信することになる．結果として動画像の品質劣化を促し，ユーザ満足度(QoE: Quality of Experience)に影響を及ぼす．また，ゲーム動画では画面に映るキャラクタが背景に紛れるのを防ぐため，テクスチャを強調する方法をとる場合があるが，解像度が低い場合はぼやけてしまうため視認性を低下させてしまう．

そこで，我々は動画上で視聴者が注視する関心領域を推定し，注視点から離れた領域の解像度は知覚できない程度に削減を行うことで動画の伝送量を抑える手法を提案する．

2 関連研究

岡田らは視線追跡器によって計測された様々な映像に対する視聴者の注視点を利用して Foveated Imaging (FI)処理を施した[2]．これは人間の視野の中心では最高解像度で物体をとらえるが，中心から離れると解像度が次第に低下するという特性を利用したものである．この手法により QoE を落とすことなく伝送量削減を実現した．しかし，エンコード前に注視点情報を何らかの通信路を経由して受け取る必要があるため，それが遅延すると注視点が移動した際にうまくフィードバックできず，視聴者が解像度の劣化を感じてしまう．

我々の先行研究では，物体検出器(YOLOv5)を用いて関心領域を特定し，領域に応じて解像度を変更する伝送量削減手法[3]を提案した．この手法ではゲームプレイ動画に対して「キャラクタが映る領域」とそれ以外の解像度を下げた「背景領域」に分割し，これら 2 つを送信する．そして，視聴者側で合成することで高効率動画を作成している．

Transmission Data Reduction considering User's Interest Regions for Efficient Game Video Distribution
† Junya Komamaki・Tokyo Denki University
‡ Katsuto Nakajima・Tokyo Denki University

実験の結果， YOLOv5x による関心領域の検出では F 値が 89%(IoU=0.4)となり，解像度変更によって生成された動画は元動画のデータ量から 0.14~0.18 倍に圧縮することができた．しかし，満足度についての主観評価では従来の低解像動画と大差ない結果しか得られなかった．原因として，キャラクタが出現するまでその領域が低解像度に設定されていることが挙げられる．

3 顕著性を考慮した圧縮動画生成

視聴者の注視点を推定する深層学習モデルを用いて FI 処理を施すことで視線情報の送受信を不要とする．具体的には，推定結果を基に生成された顕著性マップの濃度に従って解像度を 3 段階に設定することにより，キャラクタが映っているか曖昧な領域に対しても中程度の解像度に変換し，視認性低下を抑制する．処理の流れを図 1 に示す．

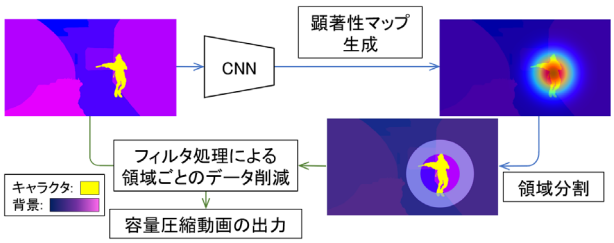

図 1 処理の流れ

以下では，視聴者の注視点(以降，関心領域と称する)の推定方法と推定した関心領域以外の情報量削減手法について詳述する．

3.1 関心領域の推定

深層学習に基づく物体検出器の特徴抽出層を可視化する技術を応用し，これを視聴者が注視するであろう関心領域と見なし，顕著性マップとして取り出す．今回，顕著性の抽出手法として Eigen Class Activation Map(EigenCAM)[4]を使用する．これは，説明可能な AI(XAI: eXplainable AI)のための技術の 1 つで，AI モデルが推論した結果の根拠を人間が解釈可能な形で出力するものであり，従来手法と比較して高速で様々なタスクに適用できる．

通常，画像認識ネットワークの畳み込み層(CNN)は認識の前段階として，複数のカーネルフィルタを用いた特徴量抽出を行うとされる．CNN の最終層からは画像内のどこに特徴が存在するかという反応の濃度を示した特徴マップが出力され，全結合層はそれを基にクラス分類等を行う．

EigenCAM ではこの特徴マップに対して特異値分解を行う．特徴マップと最大特異値に対するベクトルの積を求めることで顕著性マップを作成する．

このマップは画像認識ネットワークが物体を認識する際に根拠とした領域を可視化する．この手法による顕著性マップ$L_{Eigen-CAM}$は下記の(1)式で算出される．クラス活性化出力である式(2)の$O_{L=k}$は最終のk番目の畳み込み層に投影された入力画像Iに対する出力である．

$$L_{Eigen-CAM} = O_{L=k}V_1 \quad (1)$$

$$O_{L=k} = W_{L=k}^{T}I \quad (2)$$

本手法ではこの出力結果を視線追跡器の出力と見立てて人間が画像内である物体を認識するときに注視するであろう関心領域を推定する．

3.2 顕著性マップに基づく情報量削減手法

CNN が推定した際に生成された顕著性マップから顕著性の低い領域にガウシアンフィルタを適用する．このフィルタは画像情報の高周波成分を削減する効果があり，離散コサイン変換を用いた動画像圧縮では本来残される高周波成分を切り取ることで圧縮率の向上が見込める．

先ほどの顕著性マップから画素ごとの重みを参照し，閾値処理により画像を高品質・中品質・低品質の 3 領域に分割する．高品質の領域は平坦化処理を行わず，中品質・低品質の領域はガウシアンフィルタのカーネルサイズとσ値を変更することによってデータ削減量を調節した．

4 実験と評価

4.1 対象ゲームと学習データ

FPS ゲーム「Valorant」でのキャラクタ検出を目標とし，まず，COCO Dataset で事前学習済みの YOLOv5m に対して，CrowdHuman Dataset を用いて転移学習したモデルを作成した．次に，このモデルに対して，ゲームプレイ動画をキャプチャした画像にキャラクタ等に対してアノテーションを付与した自作データセットで再び転移学習させた．

2 つ目の転移学習時のパラメータとゲームキャラクタ等の検出精度は以下の表 1 の通りである．

表 1 学習パラメータと推定精度

入力時の解像度	1280 ×720 px
訓練用画像	934 枚
検証用画像	234 枚
テスト用画像	130 枚
mAP@0.5 (テスト)	90.0%
Recall (テスト)	86.8%

4.2 データ量の削減効果

プレイ動画の 3 つのサンプルに対して本手法で処理した場合のデータ量を，元動画，および，単純に解像度を低下させた動画のそれぞれのデータ量と比較した．結果を表 2 に示す．

FHD(1080p)の動画に提案手法を適用した結果，元動画のデータ量の約半分に圧縮することができ，HD(720p) の動画に適用した場合は，SD(360p)に低解像度化したときのデータ量と大差ない．

表 2 プレイ動画のデータ量 (MB)と圧縮率(%)

	サンプル 1	サンプル 2	サンプル 3
元動画(1080p)	183.2MB	145.8MB	158.1MB
提案手法(1080p)	50.1%	52.1%	46.2%
低解像動画(360p)	26.7%	26.8%	27.5%
提案手法(720p)	28.4%	29.8%	26.3%

4.3 動画品質の主観評価

ゲーム動画の視聴経験がある学生 21 名に対して，前節のサンプル 1, 2, 3 を視聴して貰い，画質に関するアンケート調査を行った．その結果，表 3 に示すように，全てのサンプルにおいて，全体の解像度を 360p に下げた動画よりも提案手法(720p)で生成した動画の方がキャラクタ等を視認しやすいとの評価を得た．

表 3 提案手法(720p)に対するアンケート調査結果

	サンプル 1	サンプル 2	サンプル 3
元動画(1080p)とキャラクタの動きの品質に変化が無い	66.7 %	71.4 %	71.4 %
低解像(360p)よりキャラクタの動きが分かりやすい	95.2 %	100 %	100 %
低解像(360p)より動画内の状況を把握しやすい	81.0 %	85.7 %	76.2 %

5 まとめ

本研究ではゲーム動画において視聴者がキャラクタ等に注視しやすい傾向に注目し，それ以外の領域に対してデータ圧縮しやすい処理を施すことにより視聴者の満足度を維持した動画圧縮手法を提案した．まず，独自学習を行った YOLOv5m から EigenCAM により関心領域を推定し，キャラクタ等の高関心領域は高画質のままとした．キャラクタが出現しやすい中程度の関心領域は中画質になるように平滑化フィルタ処理し，それ以外の領域は強い平滑化を行い低画質に落とした．これにより，データ量を元動画の約 0.5 倍に圧縮できた．また，ゲームプレイ動画の視聴アンケート調査では，低解像度動画と同じデータ量に圧縮した際も，本手法により生成された動画は高い評価を得た．

今後の課題として CPU によるプログラム処理速度が毎秒約 3 フレームであることから動画フレーム処理の並列化による処理速度の向上を目指す．

参考文献

[1] TwitchTracker:https://twitchtracker.com/statistics, 2022/10/9 参照.

[2] 岡田光弘, et al., "視覚特性に基づく高効率映像圧縮伝送システム," 情報処理学会論文誌 59.7 (2018): 1425-1434.

[3] 駒牧潤也, 中島克人, "キャラクタの精細度を維持しデータ通信量を抑えたゲーム配信手法," 第 84 回情報処理学会全国大会(2022).

[4] M.B.Muhammad, et al., "Eigen-cam:Class activation map using principal components," 2020 IJCNN, IEEE, pp.1-7.