

GPUによるウェーブレット変換を用いた 高精細映像の低遅延ネットワーク配信

安藤英俊[†] 望月雄太^{††}

本報告では、比較的安価に入手可能な装置を用いてフルハイビジョン映像を含む高精細映像を 100Mbps 程度のネットワーク環境においても安定的に配信する手法について述べる。映像配信の高品質化が望まれる中、特に双方向通信による遠隔講義や遠隔医療への応用を想定し、低遅延なネットワーク配信を実現した。具体的には GPU 上での離散ウェーブレット変換と離散化・符号化により高速かつ低遅延で高品位な画像圧縮を行った。これにより映像品質の制御を容易にし、ネットワークトラフィックの安定化も実現した。

Low-Latency Full HD Video Streaming using Wavelet-based Compression on GPU

Hidetoshi Ando[†] and Yuta Mochizuki^{††}

This report describes a method for streaming High Definition Video over 100Mbps network stably, only using fairly cheap devices. Very low-latency transmission of high quality video was achieved, which is especially needed for Distance Learning, Telemedicine and Telecare. We used GPU for fast discrete wavelet transform and coding for compressing high quality images. Using wavelet-based compression also achieved the ease of image quality control and stable network traffic.

1. はじめに

近年インターネット上での映像配信は様々な形で利用され、定着しつつある。動画投稿サイトからの映像ダウンロードのようなオンデマンド型のものから、イベントの片方向ライブ配信、そしてテレビ会議のように双方向で対話的なシステムまで、様々な分野で利用可能となっている。特にここ数年ではネットワーク帯域の拡大と PC の性能向上などにより、Skype 等の無料テレビ会議ソフトを使って誰でも手軽に安価にインターネットを介したテレビ会議を楽しめるようになってきた。

このように広く普及してきた安価なテレビ会議環境では現状では映像品質があまり高くなく、個人レベルでの会話程度では問題ないものの、遠隔講義や遠隔医療などの現場での本格的な活用に耐えるほどではない。一方で Polycom 社等のハイビジョン (1280x720) 対応テレビ会議システムは数百万円以上、フルハイビジョンの映像配信システム等は一般に業務用で数百万から数千万円と非常に高価である。ただしこれらのシステムでは映像配信が非常に低遅延で行えるために、遠隔講義や遠隔医療での双方向利用に向いている。大学等の一部の遠隔講義においては MPEG-2 ベースのフルハイビジョンの映像配信システムが利用されているが、片方向で 1 秒程度の遅延が生じてしまい双方向の会話に支障をきたす場合があることが指摘されている。

また日本の遠隔講義実験においては従来 DVTS (Digital Video Transport System) [1] が広く使われてきた。これは IEEE1394 規格のインタフェースをもつ DV 規格対応ビデオカメラを PC につなぐことで 30Mbps 程度の帯域を使って SD 解像度の映像を比較的遅延で配信可能である。また民生用の DV ビデオカメラとネットワークにつながった PC を IEEE1394 ケーブルでつなぎさえすれば比較的安価に利用可能であるために、初等教育レベルでの実験も多数行われてきた。DVTS では映像・音声の圧縮をカメラに行わせ、IEEE1394 パケットを PC で受け取り UDP ダイアグラムや TCP パケットに変換してネットワークへ流し、受信側では PC 上で表示したり IEEE1394 パケットに戻して DV カメラやビデオデッキを使って伸張・テレビ出力する仕組みをとる。このためソフトウェア自体は非常にシンプルな構成となっており、無料版 DVTS でも基本的な機能は一通り揃っている。その一方で最近では地上デジタル放送が確実に普及し、テレビもビデオカメラもハイビジョン対応となり、DV カメラは姿を消しつつある。そこで次の段階としては比較的安価な装置の組み合わせだけでフルハイビジョン映像をインターネット上で双方向に低遅延に配信する仕組みが求められている。

[†] 山梨大学
University of Yamanashi
^{††} 富士通 株式会社
Fujitsu Limited

2. フル HD 映像の低遅延配信

DVTS は DV 規格と DV カメラ, そして IEEE1394 インタフェースをうまく活用したために普及したが, それらと共に消滅する定めにある. DV カメラは今後徐々に市場から姿を消し, IEEE1394 インタフェースを推進していた SONY や Apple ですら自社 PC に同インタフェースを搭載しなくなった. DV 規格のハイビジョン版ともいえる HDV 規格(1440x1080)対応カメラは一部の業務用で残るのみとなっており, 将来のフル HD 映像を配信する枠組みとして DVTS を使うことはできない. そこで我々はまったく異なる手法によって, 比較的入手が容易で安価な機器を組み合わせることでネットワーク上でフル HD 映像を低遅延に配信するシステムを構築することとする.

まず現状でフル HD 映像を記録・配信する規格としてはデジタル放送等で利用されている MPEG2-TS や一部のビデオカメラで採用されている AVCHD 等があるが, 民生用カメラで映像を低遅延に取り出す仕組みとはなっていないために映像のキャプチャから配信まで一貫して利用することは出来ない. そこでフル HD 映像のキャプチャとしては現段階で唯一利用可能なデジタルインタフェースとして HDMI(High-Definition Multimedia Interface)を用いることとする. 多くの民生用 HD ビデオカメラが HDMI 出力に対応しているが, 現状では PC 側で HDMI 入力のキャプチャに対応しているデバイスの選択肢はさほどなく, 今回の実験では Intensity Pro を用いた. 将来的には最高 5Gbps のデータ転送が可能な USB3.0 などがビデオカメラと PC をつなぐ標準的なインタフェースとなることを期待している. フル HD 映像のキャプチャには 1.5Gbps 程度の帯域が必要であり, 普及している USB2.0 では帯域不足であるが, USB3.0 が普及すればそれに対応した高解像度の Web カメラ等もコストパフォーマンスに優れた有力な選択肢となる可能性がある.

1920x1080 の解像度を持つフル HD 画像を毎秒 30 枚程度送出するためにはネットワークの帯域が 1.5Gbps 程度必要となり, これは現状高速ブロードバンド接続として普及している 100Mbps 程度の帯域を大幅に超えてしまうために, 映像の圧縮が必要となる. 我々は現状で比較的安価に利用可能な 100Mbps 以下のネットワーク帯域でのフル HD 映像の配信を一つの基準とする. そこでキャプチャした HD 画像を, 比較的安価に入手可能な装置を用いて毎秒 30 回の頻度で低遅延に圧縮・符号化してネットワークに送出する必要がある. これを実現するための装置として GPU を用いることとした. この GPU 上での離散ウェーブレット変換(DWT)を用いることでフル HD 映像を低遅延に圧縮し, ネットワーク上に配信する手法を考案した. 図 1 にシステム構成図を示す. 映像送信側での処理手順としては, HDMI 経由でフル HD 映像の 1 フレームを PC にキャプチャし, それを GPU 側に YCrCb 変換をしながら転送, GPU 上で DWT を行い周波数成分に分解, 高周波成分の量子化を行い CPU に戻し, 符号化とネットワーク送信を行う. これらのプロセスを以下で詳しく説明する.

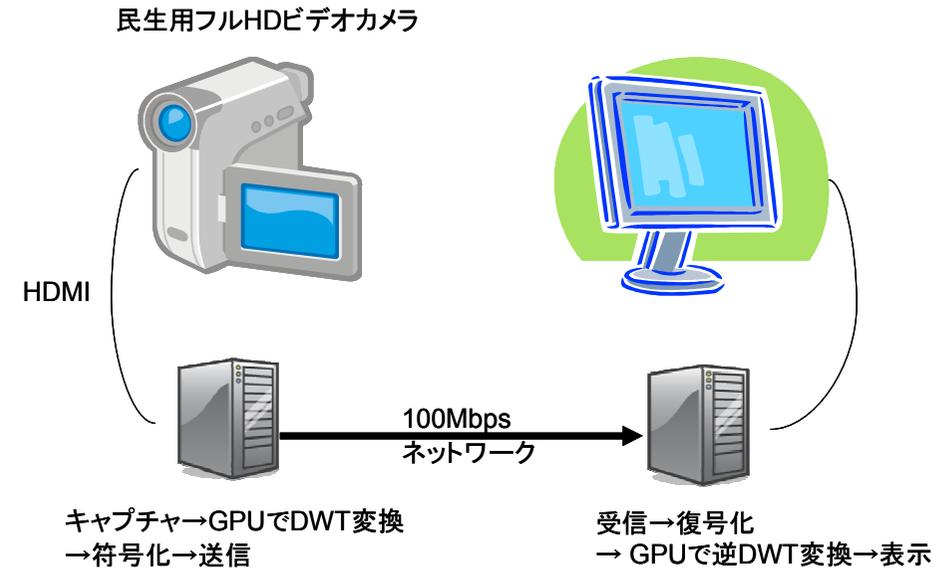


図 1 映像配信システムの構成

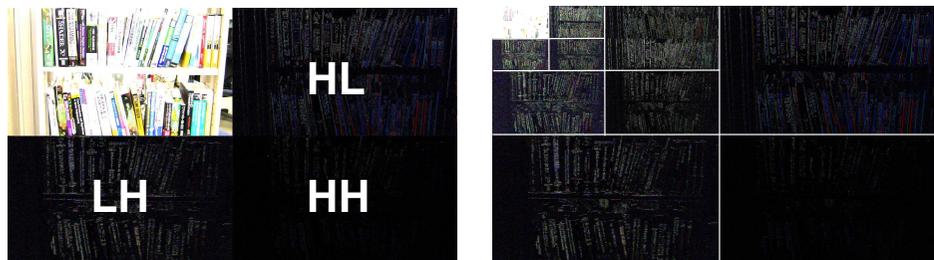
2.1 GPU 上での離散ウェーブレット変換(DWT)

まずカメラからキャプチャされた 1 フレーム分の画像は RGB 形式でメモリ上に格納される. 今回の試験的実装では Windows 上で DirectShow を利用して Intensity Pro で HDMI 入力からキャプチャした. その上で人間の視覚特性を考慮して YCrCb 変換し, 色差成分の解像度を輝度成分の解像度の半分に間引きする. そしてこれを GPU に転送して DWT を行う.

ウェーブレット変換自体は JPEG2000 等でも用いられており, 同等の品質に圧縮した場合に JPEG のベースとする離散コサイン変換(DCT)よりも高い品質を実現可能であるといわれている. このため DWT を映像圧縮に利用する研究も試みられているが, DCT と比較して計算負荷が高いことから CPU で処理した場合には低解像度の映像しかリアルタイム変換できず, また数メガバイトの作業用メモリを必要とするために専用チップ化による高速化も容易ではない. その一方で圧縮効率の高さや品質制御のしやすさなど非常に魅力的な特徴を備えた技術である. 我々は今回 GPU を用いることで高速な DWT を実現する. GPU は安価なものでも数百メガバイト程度の高速度メモリを備えており, 高度に並列な演算により高速に DWT を実行可能である.

ただし低遅延な映像配信のためには CPU のメモリ上にある 1 フレーム分の画像データを GPU 上に高速に転送して処理する必要がある。我々の実験では DirectX や OpenGL などのグラフィクス API を用いた場合にはデータ転送に時間がかかってしまい、GPU 上での DWT 実行に許される時間がかなり制限されてしまうことがわかった。そこで今回はよりオーバーヘッドの少ない手段として NVIDIA の GPU 上で利用可能な CUDA(Compute Unified Device Architecture)[2]を用いることとした。CUDA は GPU を並列計算機として利用するための環境であり、グラフィクス API よりもはるかに少ないオーバーヘッドで GPU を利用可能であるが、NVIDIA の GPU でしか利用できない。将来的には多くの GPU ベンダーがサポートする OpenCL[3]への移行が望ましい。

ウェーブレット変換としては今回は Daubechies(N=3)を基底として用い、2次元の画像に対して縦横にそれぞれ3回の変換を行った。今回対象としたフル HD 映像では縦の解像度が 1080 であるために3段階の変換がちょうどきりの良い回数であったが、任意の解像度に対してはパディング等で調整可能である。(図2参照)



(a) 1段階のDWT後

(b) 3段階のDWT後

図2 2次元DWTによる1フレーム分の画像の変換
 (高周波成分は10倍に強調)

今回は GPU 上での DWT がどの程度フル HD 映像の配信に有効であるかの基礎的な検証を行うために、映像圧縮手法としてのフレーム間圧縮は行わなかった。このため各フレームは独立に圧縮されており、動きに対しての画像劣化とは無縁になっている。

2.2 符号化

GPU 上での DWT 後には量子化と符号化を行う。Daubechies 基底を用いることによって高周波成分が小さく押さえられているため、高周波成分のゼロ値付近を削除してしまっても映像の品質に与える影響を最小限にとどめることが出来ている。また変換後のデータを直流成分および低周波成分から順に並べてシリアル化することで、任意の時点でデータストリームを打ち切ってしまうと画像の再構成がある程度可能とな

る。DWT を導入することによってこのような圧縮データのプログレッシブ化も行え、帯域制限や品質制御のための有効な手段となる。基本的に高周波成分ほど強い量子化を行っている。

量子化後の符号化としては様々な手法があるが、今回は比較的単純なランレングス符号化とハフマン符号化を併用した。高周波成分にゼロ値が頻出するために、ランレングス符号化は比較的有効な圧縮手段となる。またゼロ値以外には高周波成分の値はゼロ値を中心とする正規分布を成すことが知られているため、あらかじめ値の出現頻度が予測可能である。このため事前に用意したハフマン木を用いることで高速に効率よく圧縮が可能である。

2.3 マルチスレッド処理

符号化後はデータのネットワークへの送信に移るが、今回は Winsock を用いて UDP による通信を行った。またパケット落ち等に対する対策は特に施していない。本格的な実用化のためには符号化も含めての対策が必要である。

以上の処理はそれぞれの過程がそれなりの処理負荷を持つために、逐次的に処理した場合にはキャプチャから送信終了までに 30ms 以上の時間がかかってしまう場合が想定される。そこでこれら一連の処理を 3 スレッドに並列化することで、一連の処理としては 30ms を超えてしまっても確実に毎秒 30 フレームを処理できるように対処した。近年はノート PC の CPU も 2 コア程度は備えているため、マルチスレッド化は特に問題のない前提と考える。スレッド数は 3 であるが 2 コア CPU で十分なパフォーマンスが確認された。(図3参照)



図3 送信側処理のマルチスレッド化による効率化

映像の受信側では受信したデータから復号化を行い、GPUに転送して逆DWTをし、得られた画像をOpenGLによって画面に描画した。受信側においても、受信と復号化を一つのスレッド、GPU転送後の処理をもう一つのスレッドとすることで処理の効率化を実現した。

3. フル HD 映像配信の評価実験

以上の手法をWindows上で実装し、CPUにIntel Core 2 Duo E8500、GPUにNVIDIA GeForce GTX 260を搭載し1Gbpsのネットワークに接続された2台のPCを用いて映像の圧縮・復元処理の時間、圧縮率と品質等についての評価実験を行った。処理時間については上述のCPUおよびGPUではまったく問題なく、より低いスペックの装置でも十分に実用に耐えることがわかった。またカメラからの映像入力から圧縮・送受信・受信側での表示までの遅延は概ね3フレーム程度であり、DVTSと比較しても遜色のない程度に低く抑えられており、双方向通信によるテレビ会議の様な対話的なシステムで十分に利用可能であることも確認された。

3.1 圧縮による品質の評価

本研究では100Mbpsネットワーク環境を一つの基準とし、これ以下の帯域における圧縮率と品質の関係について評価を行った。画質の評価尺度としては広く用いられているPSNR(Peak Signal to Noise Ratio)とSSIM(Structural SIMilarity)の2つの指標について値を計測した。大まかな目安としてPSNR値は35以上であれば高品質、SSIM値は0.98以上でオリジナルと見分けがつかず、0.95程度で画質の劣化が見取れるレベルであると言われている。以下に複数の異なる環境下での画像について100Mbps、80Mbps、60Mbpsの映像帯域を想定した圧縮を行い、品質の評価をおこなった結果を表1に示す。以下ではこの表を参照しながら考察を行う。ここでビデオカメラ側の映像出力はプログレッシブモードとしているために、コーミング等の悪影響は出ていない。

表1 様々な画像に対する圧縮と品質の評価結果

指標	100Mbps 圧縮		80Mbps 圧縮		60Mbps 圧縮	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
図 4	38.2	0.968	37.4	0.965	35.8	0.951
図 5	40.8	0.979	40.8	0.977	39.3	0.979
図 6	30.98	0.981	28.88	0.973	27.4	0.953
図 7	40.5	0.977	40.5	0.977	40.5	0.977



(a)入力画像



(b)入力画像拡大



(c)100Mbps 圧縮



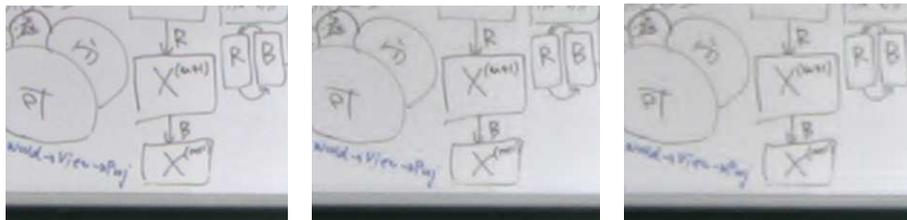
(d)60Mbps 圧縮

図4 屋外風景と圧縮結果

図4は屋外の風景で、山の木々や枝に金属ネットなど非常に細かく高周波成分の多い画像である。こういった画像は圧縮すると徐々に高周波成分がカットされてぼやけた画像となりやすい。PSNR値は60Mbpsまで35以上と高品質であるがSSIM値は圧縮と共に低下している。拡大画像(b)~(c)を見ると枝や電線等が徐々にぼやけてゆく様子が確認できる。しかしJPEG画像のようなブロックノイズやモスキートノイズが生じにくいのがDWTによる圧縮の特徴で、良好な傾向を示しているといえる。



(a)入力画像



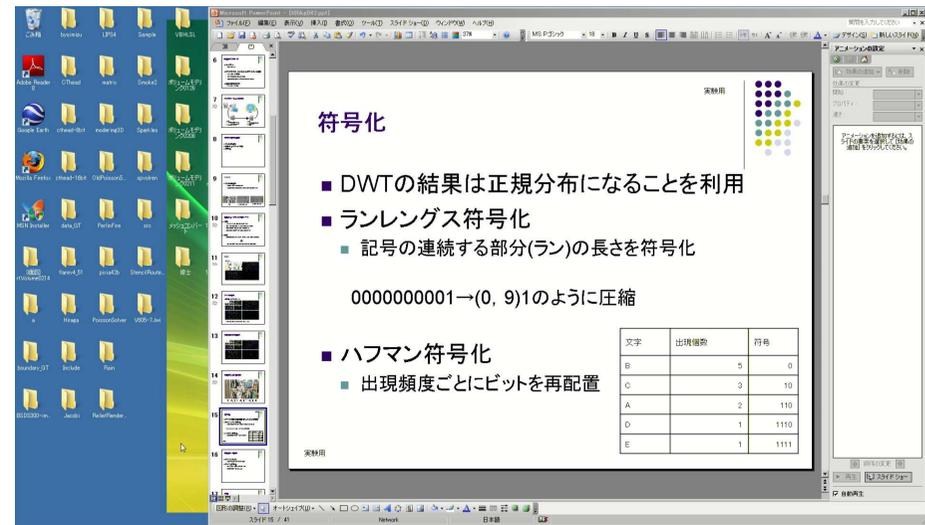
(b)入力画像拡大

(c)100Mbps 圧縮

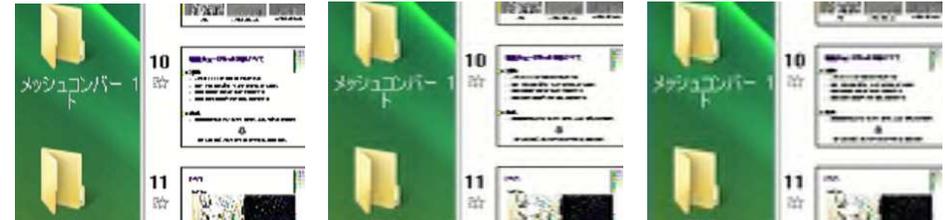
(d)60Mbps 圧縮

図5 屋内風景と圧縮結果

図5は、とある研究室の内部の様子を撮影した画像である。ホワイトボード上の手書きの図や窓のブラインド周辺に高周波成分が存在する一方で、黒板等には高周波成分がほぼ存在しない。高周波成分の多いホワイトボード上の一部を拡大して圧縮による劣化を観察すると、マーカーによる線が圧縮と共にぼやけていき、細かい文字の判別が困難になる様子が確認できる。しかしPSNR値もSSIM値も高い値を保っており、全体としては良好な結果となっている。



(a)入力画像



(b)入力画像拡大

(c)100Mbps 圧縮

(d)60Mbps 圧縮

図6 デスクトップ画面と圧縮結果

図6はフルHD解像度のPCのデスクトップ画面の配信の様子である。講義や打合せの際にはPC画面の転送が有用な場合があると評価した。結果としてPSNR値もSSIM値も低く、圧縮による画像の劣化が顕著である。特に60Mbpsの品質では細かい文字の判別が難しい。この種の人工的な画像はJPEG圧縮よりもGIF圧縮の方が良好なのと同様にDaubechies基底よりもHaar基底の方が適していると考えられる。また実用上はフレームレートを多少落としても問題のない状況であり、そちらによって画質を落とさずに帯域を低く落とすことも可能と考えられる。



図7 テレビ会議風景の例

最後に図7はテレビ会議等によく見られる風景で、壁の前の中央付近に人間が映っている状況を再現した。背景に高周波成分が非常に少ないために、圧縮しても品質劣化が起こりにくく、100Mbps から 60Mbps へと高い圧縮を行っても PSNR 値も SSIM 値も高いままである。DWT による映像圧縮は平坦な背景をもつテレビ会議等には適していることを示している。

3.2 帯域制御の安定性評価

次に様々な品質に圧縮された映像が、実際にネットワークで送受信されている際のネットワーク上のトラフィック状況を測定し、安定的な通信が維持されているかどうかを確認した。測定結果を図8に示すが、いずれの品質においても突発的なトラフィックの増減のない、非常に安定したトラフィック状況となっている。また画質を比較的高く保ったままで毎秒送信する画像の枚数を変更することでフレームレートを変化させて帯域を押さえる方法についても評価を行ったが、この場合でも非常に安定したトラフィック状況が継続的に観測できた。動きの少ない場面等においてはこういった手法も有効と考える。

この安定したトラフィックの実現には、DWT によって品質の保証がしやすいプログレッシブな圧縮が可能となったことが最大の要因であると考えられる。そういう意味でも GPU 上での DWT を用いた本研究の手法の優位性が示せたといえる。

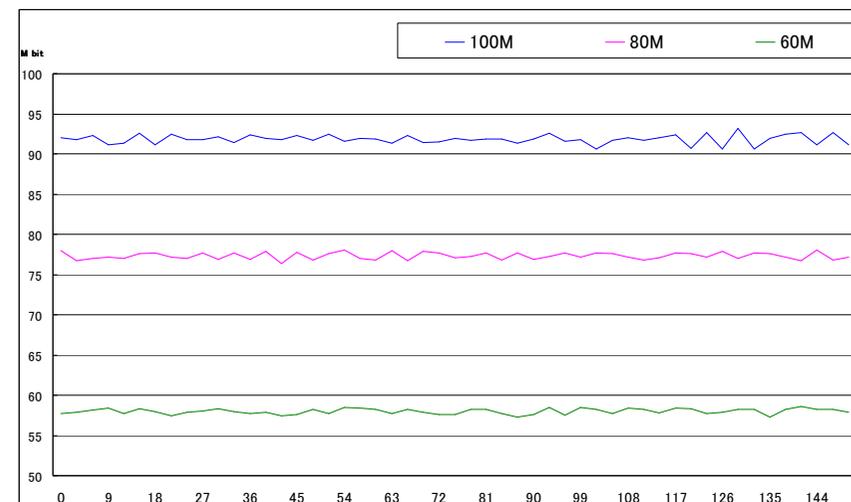


図8 様々な映像品質におけるトラフィック状況

4. おわりに

以上、GPU 上での DWT による高速画像圧縮を用いて、比較的安価な装置だけを用いてフル HD 映像を含む高解像度映像を低遅延にネットワーク配信する手法について述べた。評価実験の結果、高品質な映像を 100Mbps 程度のネットワーク環境で安定的に配信可能であることが確認された。

今後の課題としては、フレーム間圧縮による映像のさらなる圧縮、圧縮率の向上や冗長化等を可能とする先進的な符号化技術の導入、より広範囲な映像配信プロトコルへの対応などが挙げられる。またアプリケーションとして組み上げ、様々な実証実験を通じて遠隔講義・遠隔医療などへの応用可能性を今後探っていく予定である。

参考文献

- 1) Akimichi Ogawa, Katsushi Kobayashi, Kazunori Sugiura, Osamu Nakamura, Jun Murai, "Design and Implementation of DV based video over RTP", Packet Video Workshop 2000 (2000)
- 2) NVIDIA CUDA Zone, http://www.nvidia.co.jp/object/cuda_home_jp.html
- 3) OpenCL, <http://www.khronos.org/opencl/>