

# VPN環境を用いたユビキタスデータ分散処理に関する検討

甲山 絵梨奈 (指導教員：小口 正人)

## 1 はじめに

近年、ビデオカメラなどのマルチメディアデバイスの普及が進んできており、それに伴い複雑な動画処理のニーズも高まってきた。また、ネットワークの高速化により、映像データのような情報量の多いストリームデータを複数の拠点に配信することや、遠隔地のデータベースにリモートアクセスし必要な情報をネットワーク通信による遅延を気にすることなくリアルタイムに受け取ることなどが容易に実現可能である。さらに、映像データ処理技術の進歩により、複数の拠点で連携して分散処理を行うこともできる。そこで、本研究ではVPN(Virtual Private Network)を利用したユビキタスデータの機能分散処理を遠隔地で行うために、最適なネットワークシステムの構築手法について検討する。

## 2 Virtual Private Network

VPNとは、安価な公衆網において暗号化技術やユーザ認証技術により信頼性の高い通信網を構築する技術である。近年のルータには、このVPNを構築する機能や暗号化機能も含まれることが増えてきた。情報のネットワークによる共有が進む一方で、第三者による悪意ある攻撃が近年脅威となっている。VPNを用いて公衆網をより安全性の高いものとして利用する技術は、今後ますます注目を浴びると考えられる。



図 1: VPN

## 3 分散処理

分散処理とは一般に負荷を分散させるものであることが多い。これは、単独のコンピュータで処理するには高すぎる性能が要求されるような大規模な計算を、分割し並列動作させることで処理時間を短縮しようというものである。(図2)これに対し、機能分散とは機能毎に専用機に分散するような処理方式である。分散する宛先の各コンピュータは様々な異なる機能を持っており、その各機能に合った仕事を渡した方が高効率となることが多い。(図3)高速ネットワーク技術の発展により、遠隔に存在する種々のコンピュータを有機的に結びつけて機能分散処理を実現するようなことが可能になってきた。

本研究ではネットワークを利用して、各コンピュータで各々特有の処理を行うような機能分散について検討していく。具体的にはVPNを利用し、地理的に分散した複数の拠点で連携してデータ処理を行う際、どこでどのような加工処理を行う形が最適であるかなどの評価を行う。

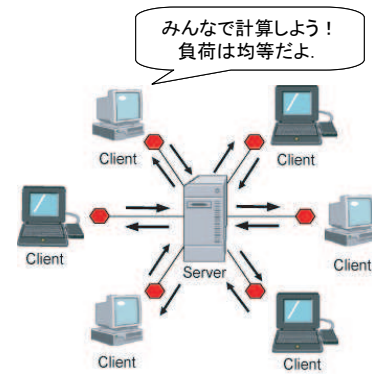


図 2: 負荷分散処理

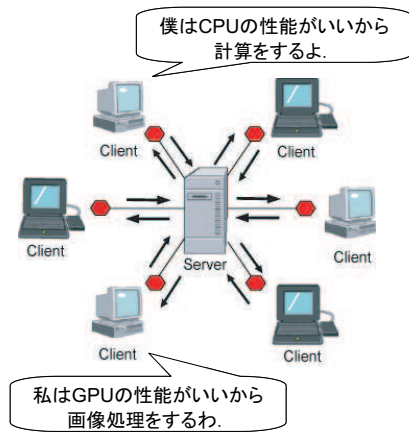


図 3: 機能分散処理

## 4 研究内容

CPU性能が向上し、コンピュータの処理速度が増すにつれて、我々の扱うデータの量もまた増大している。これに対して、情報量の多いデータを処理する為の負荷分散の研究が近年活発に行われている。

### 4.1 分散処理環境の構築手法

本研究では図4に示すように、あるノードで動画ストリームのキャプチャを行い、その映像データに対して処理を行うような環境を構築する。

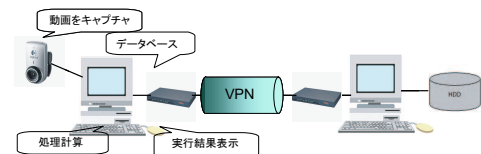


図 4: 分散処理環境の構築

画像処理ライブラリにはOpenCV[3]を使用し、USBカメラから取り込んだ映像データに処理を行う。この映像に対して、facedetectというプログラムを実行。このプログラムはOpenCVにサンプルプログラムとして付属しているものである。図5で示すように人の顔を認識して検出し、円で囲むことにより示す。

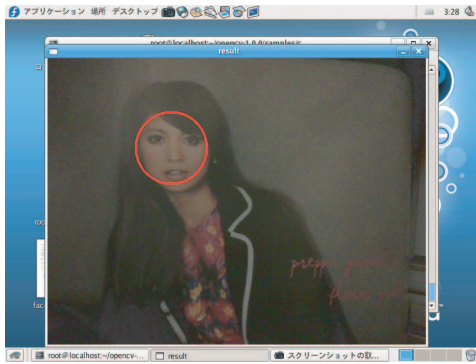


図 5: facedetect

## 4.2 USB Video Drive Class

USB Video Drive Class とは UVC と呼ばれ、Web カメラなどの映像を扱う機器を USB でパソコンに接続するための規格のことである。従来、USB カメラを接続するにはメーカーが独自のドライバをその都度開発する必要があったが、この規格の策定により、ビデオカメラ機器は1つのクラスとして定義され、OS が提供する同一クラスのドライバーを利用すればよいことになった。Linux 用の UVC も配布されており、今まで USB カメラを購入してもドライバが提供されることがほとんどなかった Linux にとっては重要な技術だと言える。

## 4.3 実験環境と実験結果

本研究では、PC として CPU は Intel Pentium III 800MHz でメモリが 640MB であるマシンと CPU が Intel Xeon 3.8GHz でメモリが 4GB であるマシンというように、性能が大きく異なるものを利用した。OS は共に Linux2.6.FC5 である。VPN ルータには富士通 Si-180[1][2]、USB カメラには Logicool 社の QCAM Deluxe for Notebooks、USB カメラのドライバには USB Video Drive Class[4]、ビデオキャプチャの API としては Video4Linux2 を使用している。USB カメラから映像ストリームデータを撮りこみながら、認識する対象である人間の顔の情報を取得するため VPN を介して他のノードにアクセスする。情報を元のノードに持ち帰り、そこで映像データに対して処理を行う。(図 6) データベース処理を行うノードがネットワー

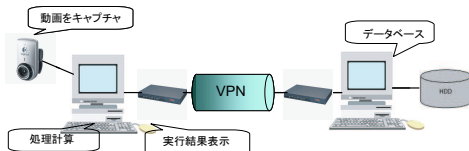


図 6: リモート DB にアクセスして処理

ク上に置かれることにより、他ノードからも容易にアクセスすることが可能となった。また、データベースを切り離して独立させたことによりデータベースのみに処理を加えることも容易となった。それぞれの機能を持つ分散環境の各ノードを VPN で有機的に結合させることにより全体として高性能な分散処理システムを構築することが可能である。その後、上記の実験に加え、データベースを所有するリモートマシンにて実

行結果を表示させた。(図 7)

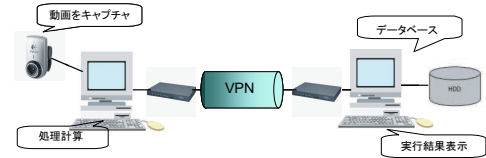


図 7: 更に結果をリモートで表示

これらの結果により、負荷の重い処理計算を性能の高いマシンで実行した場合の方が表示間隔が短くなっていることがわかる。また、データベース処理や結果表示などをネットワーク越しに分散した結果、様々な形態で同等のユビキタスデータ分散処理を実現することができた。以下に実験結果を示す。(図 8)

	左がdell・右がhpの場合 1フレームを表示にあたり		左がhp・右がdellの場合 1フレーム表示するにあたり	
	最大(ms)	最小(ms)	最大(ms)	最小(ms)
全ての処理をローカルで行う場合	1070.3	846.641	245.241	213.747
DBのみ	989.588	809.116	227.7	200.465
リモートに置く場合				
DBと結果表示をリモートで行う場合	863.068	815.061	224.581	205.885

図 8: 実験結果

## 5 まとめと今後の課題

マルチメディアデバイスの普及と共に、より情報量の多いユビキタスデータ処理について注目が集まっている。そこで、急激な高速化を遂げたネットワークを利用し、さらにその安全性を高める VPN を併用した負荷分散や機能分散処理を行う環境を構築する手法について提案した。

現在は、遠隔ノードのデータベースから情報のみを取り出し、ローカルのカメラにおいてキャプチャした映像ストリームデータに処理を施す際にその情報を利用し、また遠隔地ノードにおいて実行結果の表示も行った。その後、各ノードの役割を反転させ更なる実験を行った。今後は OpenCV の学習機能を利用し、人間の顔以外も認識できたりするよう、遠隔ノードで学習機能のプログラムも並列して実行させていきたい。その上で、ネットワークやノードの性能と機能に応じて、ユビキタスデータをどのように運んでどこで処理するかなど、VPN 環境の構築・利用手法についての検討を進めていく。

## 参考文献

- [1] 富士通ルータ Si-R 180 Web 設定事例集 ver32
- [2] 富士通ルータ Si-R 180 コマンド設定事例集 ver30
- [3] Open Source Computer Vision Library  
<http://opencvlibrary.sourceforge.net/>
- [4] Linux UVC driver and tools  
<http://linux-uvc.berlios.de/>
- [5] Linux NFS-HOWTO  
<http://nfs.sourceforge.net/nfs-howto/>