

# AI活用画像処理装置における 複数カメラのリアルタイム処理ソフトウェア開発

神戸事業所 開発部 開発第2課  
三原 隆宏

## 1. まえがき

道路管理会社は、道路ごとの交通量を計測し、その結果を基に整備計画などを立案する業務がある。これまでの交通量調査は、舗装工事を伴うループコイルを用いた自動計測や、道路に計測員を配備して手作業により交通量を計測していたが、2009年に三菱電機(株)は、交通量計測が可能な画像処理装置を備えた道路交通監視システムを開発し、交通量計測業務を自動化した。しかし、従来の画像処理装置はカメラ1台当たりの調整コストと装置単価が高いという問題があった。

今回開発した画像処理装置は、三菱電機(株)が開発したAI処理を利用することで、従来装置より調整コストを削減した。また、装置単価を下げるため、画像処理装置1台当たりのカメラ同時処理数を10台とした。

本稿では、AIを活用した画像処理装置における、複数カメラのリアルタイム処理ソフトウェア開発で取り組んだ開発要件達成のための施策について述べる。

## 2. システム概要

### 2.1 道路交通監視システム

画像処理装置を備えた道路交通監視システム概要を図1に示す。

- (1) 監視対象となる地区の道路脇に設置されたカメラは、撮影した動画を映像ストリームとしてネットワーク経由で画像処理装置に送信する。
- (2) 道路管理施設内に設置された画像処理装置は、カメラから送信されてきた映像ストリームを受信し、道路上の通行車両の車両速度、位置などの計測データを統合処理装置へ送信する。
- (3) 統合処理装置は、画像処理装置から計測データを受信し、監視対象地区の交通量や停止車両、逆走車両といった交通事象の発生有無を判断する。また、その結果を、道路管理施設内のマルチモニタや道路脇に設置した情報板などを介して、道路管理者や道路利用者に対して情報提供する。

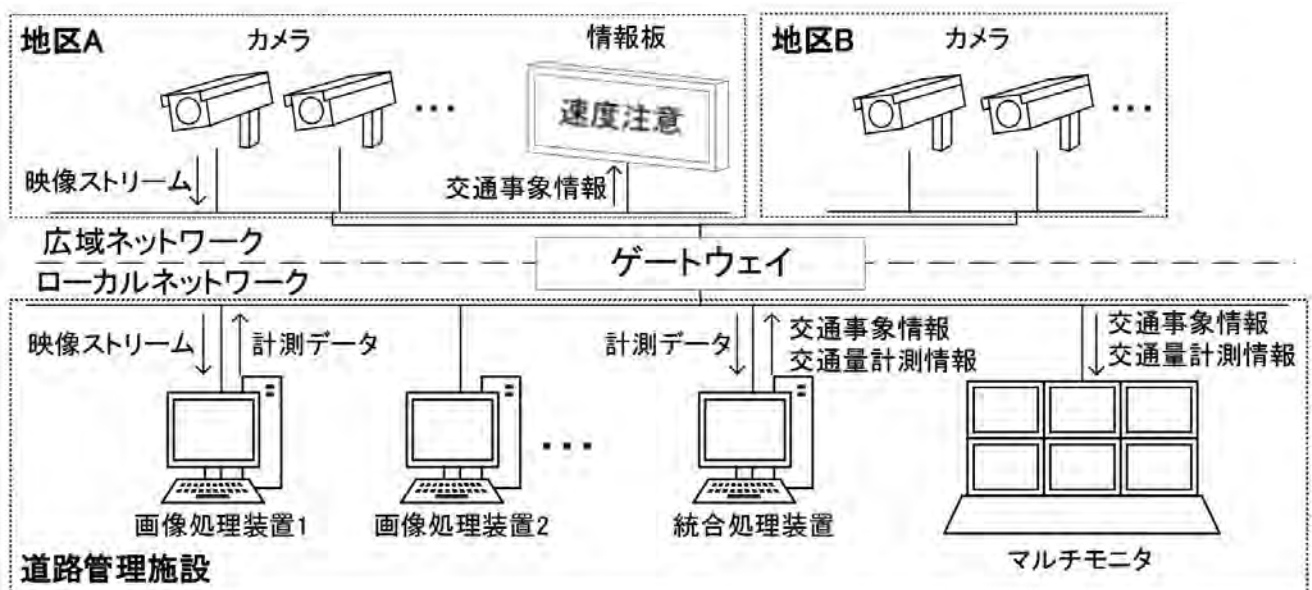


図1. 道路交通監視システム概要図

### 3. 画像処理装置概要

#### 3.1 画像処理装置

道路交通監視システム内の画像処理装置のソフトウェア機能構成を図2に示す。

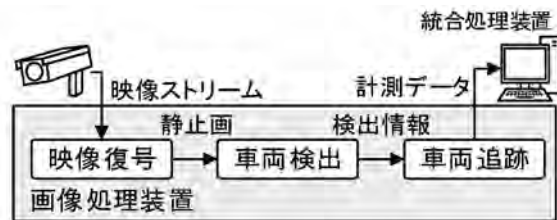


図2. 画像処理装置ソフトウェア機能構成

画像処理装置は、映像復号、車両検出、車両追跡の3つの機能で構成される。各機能における処理の役割について表1に示す。

表1. 画像処理装置 機能概要

機能名	役割
映像復号	カメラから受信した映像ストリーム(動画)を静止画に復号する。 復号結果(静止画)を車両検出機能に渡す。
車両検出	静止画から車両の位置、車種などの検出情報を出力する。 検出情報を車両追跡機能に渡す。
車両追跡	検出情報から個々の車両の追跡結果(車両速度、位置)を計測データとして出力する。 計測データを統合処理装置に送信する。

#### 3.2 画像処理装置の開発要件

画像処理装置の開発要件は以下のとおり。

- (1) カメラ1台当たり100ms周期での計測処理
- (2) 画像処理装置1台でカメラ10台分の同時処理
- (3) 処理データ欠損発生時の正しい計測処理

開発要件達成に向け、検討した結果を以下に示す。

- (1) 従来装置は、三菱電機(株)が開発した専用ハードウェア上に搭載する画像処理ソフトウェアであった。しかし、今回開発した画像処理装置は、汎用装置のWindows上に搭載する画像処理ソフトウェアである。そのため、従来装置と同等の100ms周期でのリアルタイム処理を、Windows上でどのように実現するかが課題となった。
- (2) 従来装置は、装置1台で1カメラ処理であったのに対し、今回開発する画像処理装置は、装置1台でカメラ10台

の同時処理をする。そのため、前述のカメラ1台当たり100ms周期での計測処理と合わせ、性能を達成するための同時処理実現方法が課題となった。

(3) カメラから送信される映像ストリームは、UDP/IPプロトコルのデータとなるため、回線品質により復号データの欠損が発生する。その際、計測方法を誤ると、正しい車両速度が算出されないため、対策が必要となった。

上記の検討結果から、開発要件達成のための施策を次章に述べる。

### 4. 開発要件達成のための施策

#### 4.1 GPUを活用した処理性能の確保

画像処理装置を構成する映像復号、車両検出、車両追跡の3機能について、図3に示すように、それぞれが専用プロセッサ上で独立して動作するよう設計した。

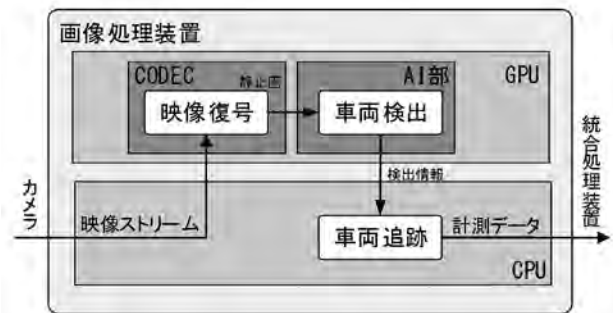


図3. 画像処理装置 機能実装構成

3つの処理機能は、それぞれ以下の特徴を持つ。

- (1) 映像復号機能  
GPU(Graphics Processing Unit)<sup>(注1)</sup>上の専用プロセッサであるコーデック(CODEC)<sup>(注2)</sup>を利用し、映像ストリームから静止画を高速に復号する。
- (2) 車両検出機能  
復号結果の静止画から、GPU上のAI処理向け専用プロセッサであるAI部で車両検出を行う。
- (3) 車両追跡機能  
検出した車両情報に対して、CPU上で車両速度などの計測処理を行う。計測結果のデータを統合処理装置に対して、100ms周期で送信する。

(注1) 数値演算処理に特化した専用プロセッサ。  
(注2) 符号化方式を使ってデータのエンコード(符号化)とデコード(復号化)を行う専用プロセッサ。

本開発では、映像復号、車両検出の処理高速化のためにGPUを活用した。GPUには、動画再生支援向けとAI処理向けの2つの専用プロセッサが搭載されており、CPUと異なり単純な演算処理を同時並行して高速に実行することができる。このため、圧縮された動画データから静止画を得る映像復号処理や、静止画を解析して画像認識するAI処理で特に効果を発揮する。動画再生支援向けの専用プロセッサはコーデックと呼ばれており、動画の符号化処理や復号化処理をハードウェア上で高速に実行できる。また、AI処理向けの専用プロセッサであるAI部では、三菱電機(株)が開発したAI車両検出処理で使用する行列演算を高速に実行できる。上記施策により、画像処理に必要な処理性能を確保した。

#### 4.2 パイプライン処理による処理並列化

画像処理装置は、統合処理装置に対して100ms周期で計測データを送信する必要がある。そのため、100ms周期で入力データとなる静止画に対する計測処理を繰り返し実施する。

図4では、1回の映像ストリームの入力に対する、GPU上のコーデックで動作する映像復号、同じくGPU上のAI部で動作する車両検出、そして、CPU上で動作する車両追跡における処理状況を示している。

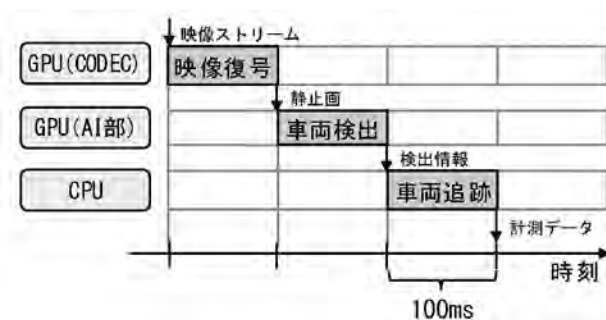


図4. CPUとGPUを用いた画像処理

図4では、GPUとCPUでは処理を行っていない期間が存在し、その期間では、別の処理が実行可能となる。そのため、図5に示すように、次の映像に対する処理が開始でき、処理の並列化を行うことが可能となる。

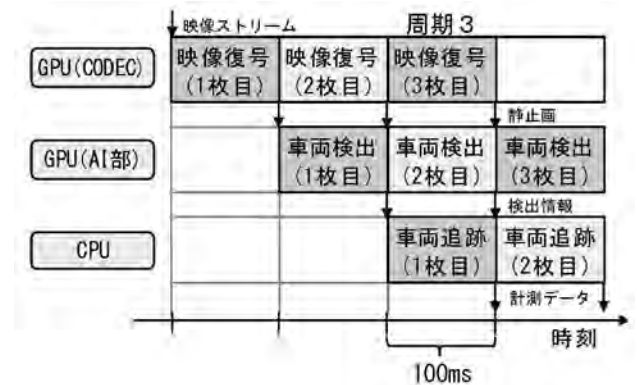


図5. CPUとGPUの処理並列化

図5の周期3では、3枚目の静止画に対する映像復号、2枚目の静止画に対する車両検出、1枚目の静止画に対する車両追跡を同時に実行している。1枚の静止画に対する処理時間は、3機能の処理時間の合算となる。1機能の処理時間を100msとした場合、処理時間の合計は300msとなるが、実行をパイプライン化することで100ms周期の計測データ送信を可能とした。

#### 4.3 一括AI処理によるGPU処理効率化

車両検出機能は、映像復号機能から静止画を受け取りGPU上で車両検出処理を行う。この車両検出では、カメラ1台当たり12msの処理時間を必要とする。カメラ10台の静止画に対し、逐次車両検出処理を行った場合の状況を図6に示す。

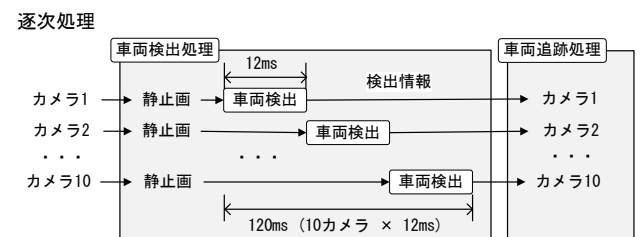


図6. 車両検出機能 逐次処理

図6では、10台のカメラに対して逐次車両検出を行うため、車両検出処理では合計120msの処理時間が必要となる。画像処理装置における各機能の処理時間は、100ms以内に完了する必要があるため、逐次処理とした場合、処理時間制約を満たさない。

GPUは同じ処理を複数回実行するより、1回の処理に対して大きなデータを実行する方が処理効率が高く、短時間での処理が可能となる。

そこで、映像復号機能から受け取った複数カメラの静止画を1枚の静止画として再構成し、三菱電機(株)が開発したAI車両検出で、再構成した1枚の静止画を一括処理する設計とした。一括処理を行った場合の処理例を図7に示す。

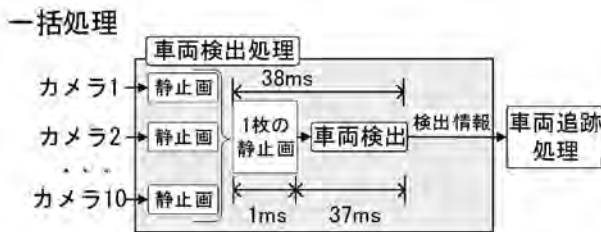


図7. 車両検出機能 一括処理

各カメラの静止画を1つの静止画に再構成する処理時間は約1msのため、複数カメラの一括処理とすることで、カメラ10台分の車両検出時間は約38msで完了でき、逐次処理より処理の効率化を実現した。

#### 4.4 GPUメモリアクセス効率化

GPUには高速アクセスが可能な専用メモリが8.0GB、専用メモリより低速アクセスとなる共有メモリが24.0GB搭載されている。車両検出処理では、図8に示すとおり、カメラ1台で画像情報を約0.1GB、学習モデル情報を約0.9GBの約1.0GBのメモリを使用する。そのため、カメラ10台分の処理に必要なメモリ量は合計10.0GBとなる。

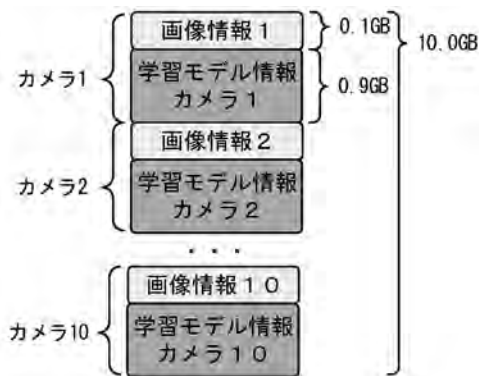


図8. 車両検出機能 必要メモリ量

カメラ10台分の処理を行う場合、必要メモリが約10.0GBとなり、高速アクセスが可能なメモリ容量の8.0GBを超える。

車両検出処理に必要な学習モデル情報は、AI処理で用いられる車両の特徴を学習したモデルデータとな

る。この学習モデル情報は、様々な角度から撮影した車両を学習したデータのため、全てのカメラで共通な学習モデル情報となる。学習モデルは、AI処理時に頻繁にアクセスが必要となるデータのため、高速アクセスが可能なメモリ領域に配置することが望ましい。

そのため、一括処理時における学習モデル情報は、全てのカメラで共通とするメモリマップとなるように変更した。図9に、変更後のメモリマップを示す。

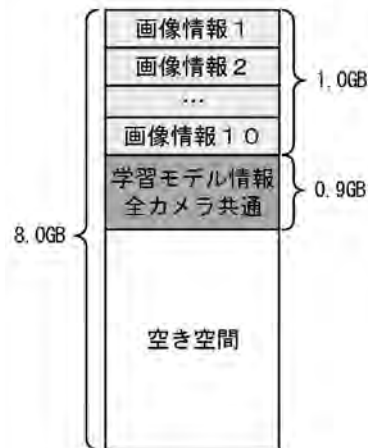


図9. 車両検出機能 メモリマップ

学習モデル情報を全カメラで共通化した情報とすることで、GPU上で必要となるメモリを合計1.9Gとすることができ、高速アクセスが可能な専用メモリの使用のみで複数カメラの車両検出処理を実現した。

#### 4.5 回線品質による処理データ欠損への対応

カメラは33ms周期で撮像と符号化を繰り返し、映像ストリームをネットワークに送信する。画像処理装置は、受信した映像ストリームを逐次復号し、100ms周期でその時点での最新の復号結果となる静止画を使用し、車両検出処理を行う。しかし、ネットワークの回線品質により、ストリームデータの欠損が発生する場合がある。欠損が発生すると、図10に示すように、画像処理装置で処理される静止画の間隔は100msではなくなり、66msや133ms間隔となる場合がある。

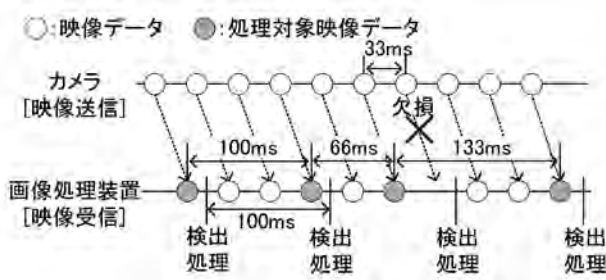


図10. 映像データ処理概要図

上記の事象が発生した場合、車両追跡機能において正確な車両速度が算出できなくなる。図11に正常状態と欠損が発生した場合の静止画差分を示す。処理データの欠損が発生すると、次に処理する静止画は、正常状態の静止画より33ms前の静止画となる。そのため、欠損が発生した場合、静止画間で見かけ上の車両移動量が少なくなる。この場合、車両追跡機能の処理間隔となる100ms周期で車両速度を算出すると、36km/hとなる。しかし、欠損が発生した場合の静止画間の正確な時間間隔は66msとなるため、車両速度は54km/hが正しい計測結果となる。

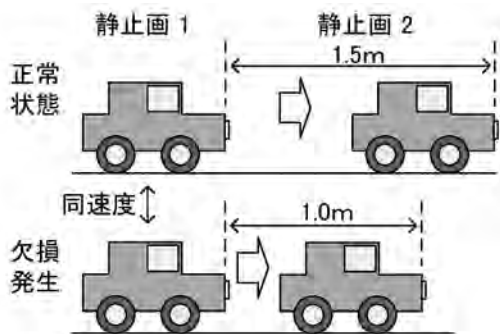


図11. 正常状態、欠損発生時の静止画差分

このような状況への対策として、カメラが映像データに付与するタイムスタンプを使用することとした。タイムスタンプはカメラ側で撮像したタイミングで付与される。処理する静止画間のタイムスタンプ差を算出することで、欠損が発生した場合でも正確な静止画間の時間間隔を得ることができる。カメラ側で付与されるタイムスタンプ情報を図12に示す。

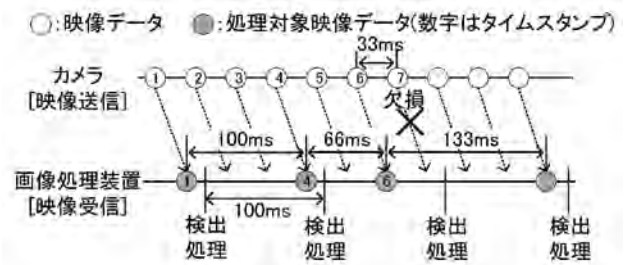


図12. 映像データに付与されるタイムスタンプ

画像処理装置の車両追跡機能において、カメラ側で付与されたタイムスタンプ間隔は100ms周期で間隔が3となるが、欠損が発生した場合は間隔が2となる場合があり、これは実時間で66ms間隔を意味する。このことから、静止画間の車両移動量とタイムスタンプから算出される時間差を基にすることで、処理データ時間間隔のばらつきが発生した場合でも正確な車両速度計測を行うことを可能とした。

## 5. 施策結果評価

画像処理装置のハードウェア仕様に基づく各機能の処理性能を、1カメラ処理時とカメラ10台一括処理時で計測した評価結果を以下の表2に示す。

表2. 画像処理 処理性能一覧

	処理時間 (ms)	
	1カメラ	10カメラ
映像復号	33	33
車両検出	12	38
車両追跡	2	2

### (1) 映像復号機能

映像復号は、GPU上の専用ハードウェアとなるコーデックで実施するため、同時処理するカメラ台数が10台となっても、処理時間は1カメラ処理時と同一時間となる。4.1節の施策を実施することで、車両検出や車両追跡処理に影響を与えない専用プロセッサ上での復号処理を実現した。

## (2) 車両検出機能

AI車両検出は、1カメラ処理時に12msの処理時間を必要としたため、4.2、4.3及び4.4節の施策を実施しない場合、8カメラの同時処理が上限となっていた。施策を実施することで、カメラ10台一括処理を38msの時間で処理を完了する。GPU処理の一括化とメモリアクセス効率化により、開発要件を満たす処理性能を達成した。

## (3) 車両追跡機能

車両追跡は、CPU上での処理となるが、計算量が少ないため、カメラ10台分の車両追跡処理を行っても処理時間の増加は発生しない。4.2節の施策により複数カメラの100ms周期の計測データ出力、4.5節の施策により正しい車両速度計測処理を実現した。

## 6. 成果と今後の展望

今回開発したAI活用画像処理装置は、カメラ10台に対して画像処理を同時に行い、車両計測を100ms周期でリアルタイム処理することを可能とした。

また、本稿で提示した施策は、今後のカメラ同時処理数の増加を見越した拡張性のある設計としたため、現状のハードウェア仕様と処理性能評価結果から、最大20台のカメラに対する同時処理が可能となっている。そのため、今後入札予定の工事における装置コストの削減にも大きく貢献できる。

## 7. むすび

本稿で紹介した取り組みで、GPU活用技術、並列化処理設計による高度なリアルタイム処理を実現することができた。また、今回利用しなかったコーデックの機能である符号化を利用することで、交通事象の映像蓄積や、車両検出や速度計測状況を映像ストリームとして配信し、遠隔監視も対応可能となる。画像処理装置開発において、常に最新の技術を導入し、顧客の高いニーズに応じていく所存である。

最後に、本開発に当たり技術的な助言をはじめとして、様々な面でサポートしていただいた関係者の方々に深く感謝申し上げます。

### 執筆者紹介



三原 隆宏                      ミハラ タカヒロ  
1998年入社。主に広域監視制御システムのソフトウェア開発に従事。現在、神戸事業所開発部開発第2課。