

リアルタイムアナグリフ動画像生成

小堀 由貴* 白川 有莉*

石川 千里** 高田 雅美** 城 和貴**

*奈良女子大学理学部情報科学科

**奈良女子大学大学院人間文化研究科情報科学専攻

近年、映画やゲーム、アートから遊園地のアトラクションに至るまでの実にさまざまな分野で三次元可視化技術が利用されている。本稿では、そのような可視化技術のなかでも特に立体視に注目し、その実現のために2台のUSBカメラから撮影した映像をOpenCVを用いてリアルタイムでアナグリフ映像に変換して出力することを試みる。またOpenCVによる処理速度低下の欠点を克服するために、映像(動画)や音声を扱うためのフレームワークであるDirectShowをキャプチャ部分にのみ取り入れて、変換処理の高速化を図る。

Real Time Anaglyph Moving Image Generation

Yuki Kohori*, Yuri Shirakawa*

Chisato Ishikawa**, Masami Takata**, Kazuki Joe**

* Information and Computer sciences Department, Nara Women's University

**Graduate School of Humanities and Sciences, Nara Women's University

In late years three dimensions are visible, and technology is used in really various fields before reaching the attraction of the amusement park by a movie and a game, art. In this paper, it looks like it: is visible, and pay attention to stereoscopic vision in particular in technology, and try what use OpenCV, and convert the picture which photographed from two USB cameras for the realization into a Anaglyph picture in real time, and output. In addition, I take in DirectShow that is a framework to handle a picture (an animation) and a sound only on a capture part to overcome the fault of the transaction speed fall by OpenCV and plan speedup of the conversion processing.

1. はじめに

近年、インターネットのブロードバンド化により、youtubeのようなストリーミング動画を誰でも簡単に配信できるようになった。また、野球中継や生放送番組のリアルタイム配信はもはや日常茶飯事である。また、3D動画は、映画や遊園地のアトラクションでの公開をよく目にする。

そこで本研究では、そういった最先端技術のうち特に3次元映像シミュレーションに注目してみる。シミュレーションに用いられている3次元映像技術の例

として、液晶シャッターメガネで立体視するといったものがあげられる。しかしながら、液晶シャッターメガネは非常に高価であるため、誰もが利用できる物とは言えない。そこで本稿では、極めて低コストかつ両眼視差により誰でも容易に3次元立体映像を楽しむことができるアナグリフ映像を、2台のUSBカメラを用いて撮影したもからOpenCVとDirectShowを駆使してリアルタイムで生成することを試みる。

2. アナグリフ動画の生成

2.1 アナグリフ画像とは

1つの対象物を、2台のカメラで撮影する。2台のカメラは対象物から同じ距離の場所で左右に数センチ離す。これにより2台のカメラから視差が発生した映像が得られる。左右の映像に、それぞれ赤と青のフィルタをかけ合成し、赤青メガネで分離して見ることにより視差が再現され、立体に見えるというのがアナグリフ画像である[1]。

その時のイメージを以下の図1に示す。

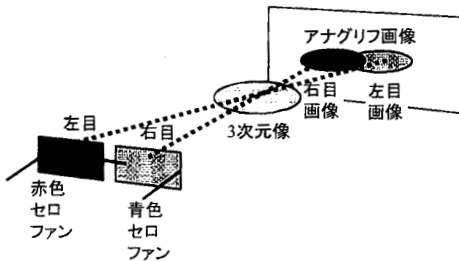


図1 アナグリフ画像が立体に見える時のイメージ

2.2 アナグリフ動画生成プログラムの構築

開発環境は以下のようにになっている。

<開発環境>

- Windows XP
- Visual C++ 2005 Express Edition [2]
- OpenCV(Intel Open Source Computer Vision Library) [3]
- DirectShow

OpenCVとは、Intel社がオープンソースで公開している画像処理に関する関数が詰め込まれたC/C++用ライブラリであり、複数のOSに対応している[4]。

DirectShowとは、Microsoft社のWindows用マルチメディア拡張API群であるDirectXに含まれるAPIの一つである。

DirectShowは基本的にBGRのビットマップ形式を受け渡しするため、同じくBGRで扱うOpenCVにおいて利用するためには、IplImage形式に変換して読み込む必要がある[5]。

アナグリフ作成にあたりプログラムの流れについて以下に示す。

まず、cvCreateCameraCapture()でカメラの初期化をし、cvNamedWindow()で表示ウィンドウの準備をする。

次にcvQueryFrame()で2台のUSBカメラからフレーム画像をそれぞれ1枚ずつ取り出す。取り出した左右の画像にそれぞれ赤と青のフィルタをかけて合成しcvShowImage()で画像を表示する。この合成手順の詳細については、2.2.1節において述べる。

Escキーが押されたらcvReleaseImage()で確保されたメモリを開放して終了する。

図2にプログラムの流れを示す。

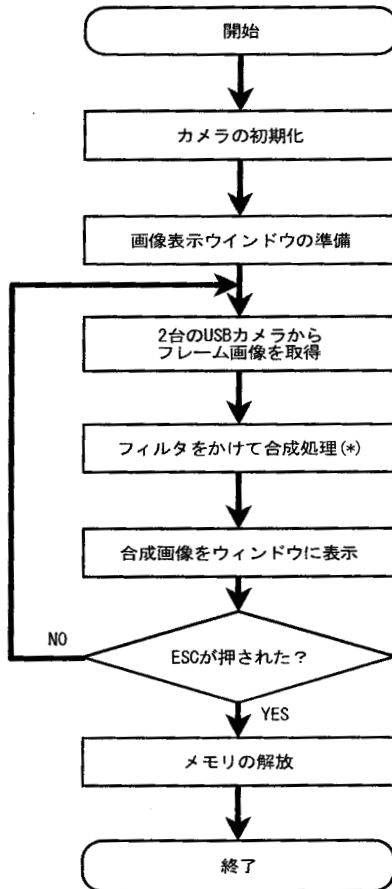


図2 プログラムの流れ

2.2.1 フィルタ適用と合成処理

図2の(*)部について説明する。ここでは2台のUSBカメラを使い、リアルタイムで赤青のフィルタをかけ、合成を行う。処理の流れを以下に示す。

- ① 左画像の各画素のGBを最大値にする
- ② 右画像の各画素のRを最大値にする
- ③ 出力画像バッファのRに②を、GBに①を入れる

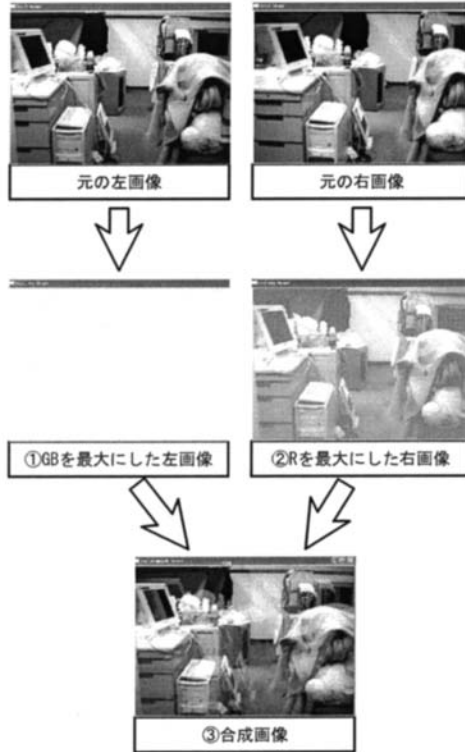


図3 アナグリフ合成処理の流れ

3. DirectShowによる高速化

OpenCVでキャプチャした結果、1フレームあたり70~90msであった。すなわち2台のカメラからの取り込みは140~180ms、フレームレート換算すると1秒あたり12~13フレームとなる。これは一般的なテレビ(30fps)に比べると大幅に処理時間が遅い。

そこで、動画や音声の処理のためのフレームワークであるDirectShowを用いることでキャプチャ部分の高速化を図る。

本手法では、カメラ画像をOpenCVでキャプチャしていた部分をDirectShow経由で行う。その後はOpenCVを用いて動画化する。

4. 実験

アナグリフ動画のリアルタイム生成実験について述べる。本実験では、実行PCとして1.6GHzのPentium Dual-Core、1GBのメモリを搭載したものをを用いる。USBカメラとして解像度640×480(約31万画素)、CMS-V20SETSVを使用する。このUSBカメラの上記PCとの接続における最大フレームレートは30fpsである。

DirectShowを使用するために、本稿ではMicrosoft社のDirectX SDK(June2006)及びDirectX 8.0[5]を用いる。

3.1 手順

DirectSnowを使用したキャプチャおよびアナグリフ変換処理について説明する。

まず、USBカメラからのBMP情報を取得し、バッファを確保しBMPデータを割り当てる。次にOpenCVの画像形式(IPL形式のimageデータ)を用意しBMPデータをコピーする。後はOpenCVでアナグリフ処理する。

4.2 結果

OpenCVでキャプチャする時間はカメラ1台につき約80ms、DirectShowでは約12ms、合成処理は共に20msとなった。

実験結果を表1に示す。

	1台目のカメラからとってき きて1枚メモリにコピーする 時間	2台目のカメラからとってき きて1枚メモリにコピーする 時間	合成 時間
OpenCV	約80ms	約80ms	約20ms
DirectShow	約12ms	約12ms	約20ms

表1 実験結果

5. 考察

図4は、2台のカメラそれぞれからとってきた画像をメモリにコピーするまでの時間と、アナグリフに合成する処理時間をOpenCVまたはDirectShowでキャプチャ

ャする場合それぞれで処理時間合計をグラフ化したものである。

図4よりOpenCVでの1台のカメラでのキャプチャには約80msかかっており、合成処理には約20msかかることから、それぞれのカメラから画像を取り出し合成するのに合計約180msとなり、5.5fpsであることに対し、DirectShowでのキャプチャは約12msあり、合成処理の速度は同じなため、合計は約44msとなり、22.7fpsとなる。

このことから、OpenCVよりもDirectShowでキャプチャした時の方がキャプチャ速度が約7倍改善されたことがわかる。

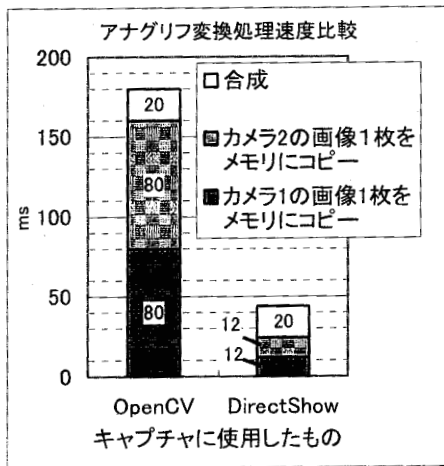


図4 アナグリフ変換処理速度比較

6. おわりに

本研究では、数ある立体映像化技術のうちのひとつであるアナグリフ画像に注目した。2台のUSBカメラで撮影した映像にOpenCVを用いてリアルタイムでアナグリフ映像に変換して出力し、さらにDirectShowの関数を利用することで、その結果22.7fpsでのアナグリフ動画生成に成功した。

今後は、人によって赤が強く見える人や青が強く見える人などさまざまなので、誰もが確実に立体視できるように、見る人に応じたフィルタの色補正ができるようにしたい。

さらにこの動画をflvファイル形式に変換して、ストリーミングサーバを開発し、ライブストリーミング配信を実現させることを目指す。

7. 参考文献

- [1] 立体写真の見方/アナグリフ
<http://www.stereoeye.jp/howto/anaglyph.html>
- [2] 林晴比古, 明快入門 Visual C++ 2005, 2006
- [3] 奈良先端科学技術大学院大学 OpenCVプログラミングブック制作チーム, OpenCV プログラミングブック, 2007
- [4] Open Source Computer Vision Library
<http://www.intel.com/technology/computing/opencv/>
- [5] Microsoft DirectX Developer Center
<http://www.microsoft.com/japan/msdn/directx/>