

不可視映像提示技法「Magic Light Display」における FPGA を用いたフレーム同期高速化による視認性向上

五島慶人[†] 流石勇輝[†] 橋本直[†]

概要：我々はこれまでの研究において、不可視映像提示技法「Magic Light Display」を提案した。本技法では、ディスプレイ上に外部から強い光を照射することで、その光源を持つ観察者にのみ、不可視な映像を視認させることができる。さらに、時分割制御を用いることで、光源ごとに異なる映像を提示することも可能である。初期の実装では、映像と外部光源の同期制御に PC・マイコン間の USB シリアル通信を用いていたため、提示可能な映像のフレームレートが低く、視認される映像の点滅感が強いという問題があった。また、同期ずれに起因するちらつきも顕著であった。そこで本研究では、FPGA (Field Programmable Gate Array) を用いて映像信号から同期信号を生成することにより、従来の USB シリアル通信方式よりも高速かつ安定な光源制御を可能とする実装方法を提案する。本論文では、提案システムの概要を説明するとともに、試作した装置を用いた動作検証の結果を報告する。

1. はじめに

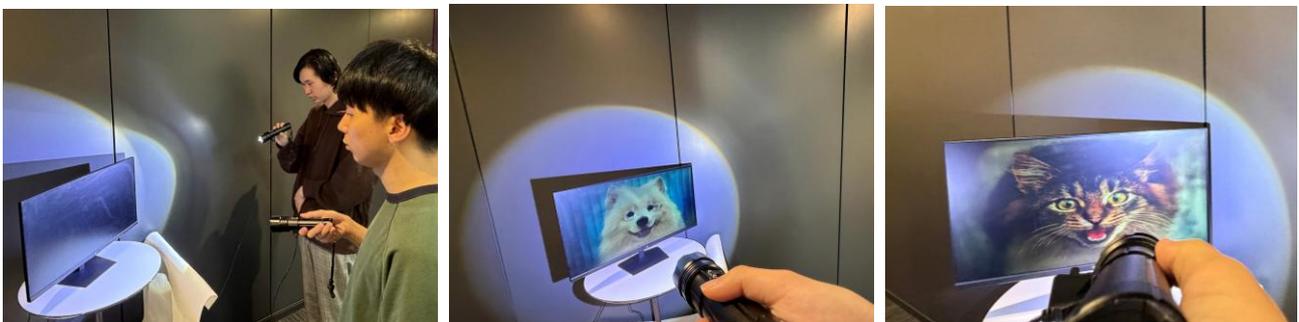
我々が提案した「Magic Light Display」は、同一ディスプレイ上で映像の隠蔽と多重化を同時に行う手法である[1]。図 1 に示すように、光源（懐中電灯）を持たない観察者には何も見えないが、光源を持つ複数の観察者は同時に映像を視認でき、かつ光源ごとに異なる映像を知覚できる。この現象は、バックライト部分を再帰性反射材で置き換えた液晶ディスプレイ上に、複数の映像を時分割で表示し、その切り替えに同期して観察者の持つ光源を点滅させることで実現している。

Magic Light Display の初期の実装では、PC と光源制御用のマイコンを USB シリアル通信 (115200baud) で接続し、PC 上で時分割映像提示を行うプログラムにおいて、フレーム描画処理を記述する関数内から光源制御指令を送信する方式を採っていた。しかし、描画関数内で指令を送信しても、そのフレームが実際にディスプレイ上に表示される時刻と、マイコンが制御信号を受信して光源を駆動する時刻とが一致する保証はなく、通信および表示系に起因する遅

延やジッタの影響を受けやすかった。さらに、PC 側の描画処理は OS や GPU のスケジューリングの影響を受けるため、フレーム周期が一定とならず、描画タイミングのばらつきが生じるほか、状況によってはフレームドロップが発生する場合もあった。その結果、光源制御と映像提示の対応関係がフレーム単位で崩れやすく、同期ずれによる映像の混濁やちらつきが顕在化した。

これを緩和するため、初期の実装ではディスプレイを 30 fps で駆動しつつ、画像 A と画像 B の切り替え境界で光源が発光した際に映像が混濁することを避ける目的で、画像 A → 黒画像 → 画像 B → 黒画像 → … という提示手順を採用した。このように各映像の間に黒画像を挿入することで混濁は抑制できたものの、実効的な提示速度は 7.5 fps にとどまっていた。このため、視認される映像には点滅感が強く生じた。また、描画周期の不安定さに起因してフレームドロップが発生した場合には、意図しないフレームで光源が駆動され、光源と対応しない画像が瞬時的に表示されることもあった。

そこで本研究では、提示フレームごとに左上隅 1 ピクセ



(a) 光源を持たない観察者の視点

(b) 光源 A を持つ観察者の視点

(c) 光源 B を持つ観察者の視点

図 1 動作の様子

ルに色による識別情報を埋め込み、PC から出力された映像信号を、FPGA のプログラマブルロジック (Programmable Logic) 上に実装した回路によって解析することで、実際に出力されたフレームに同期した光源制御信号を生成する実装方法を提案する。本方式では、PC 側の描画処理や OS のスケジューリングに依存せず、表示信号そのものを基準として光源制御を行うため、高フレームレートでの安定した提示が可能となり、視認映像における点滅感は解消される。また、出力されたフレーム内容に基づいて光源制御が行われるため、PC 側でフレームドロップが発生した場合であっても、提示フレームと光源の対応関係が崩れることはない。

本論文では、提案する実装方法の詳細を述べるとともに、試作装置を用いて行った簡易的な検証結果について報告する。

2. 提案手法

2.1 システムの概要

システム構成を図 2 に示す。本システムは、液晶ディスプレイ、PC、FPGA 搭載開発ボード、光源、および光源制御回路から構成される。

液晶ディスプレイには、液晶パネル背面に内蔵されたバックライトモジュールを除去し、代わりに再帰性反射材を配置する改造を施している。この構成により、液晶ディスプレイ単体では表示映像を視認できないが、外部から強い光を照射した場合には、再帰反射によって光源方向からのみ表示映像を視認することが可能となる。

PC では、画面描画を行うプログラムを実行する。本プログラムは、複数の映像をフレーム単位で時分割に切り替えながら描画する。このとき、映像の混濁を抑制するために黒画像の挿入する。例えば、2 枚の静止画 (画像 A および画像 B) を提示する場合には、1 フレームごとに画像 A → 黒画像 → 画像 B → 黒画像 → … の順で交互に描画を行う。さらに、映像信号中の各フレームがどの映像を表示しているかを瞬時に識別可能とするため、各フレームの第 1 ピクセル (左上隅のピクセル) に、RGB 値によって表現される識別情報を埋め込む。例えば、画像 A に対応するフレームでは第 1 ピクセルを (255, 0, 0)、画像 B に対応するフレームでは (0, 0, 255) とすることで、提示内容をフレーム単位で区別できるようにする。

FPGA 搭載開発ボードでは、PC から出力された映像信号を入力として受け取り、各フレームの第 1 ピクセルを読み取ることで、現在提示されている映像の識別を行う。その識別結果に基づき、提示画像に対応する光源のみが点灯するよう、光源内部の駆動回路に対して同期信号を出力する。また、FPGA 搭載開発ボードは、入力された映像信号をそのままパススルーし、液晶ディスプレイへ出力する。

光源制御における信号のタイミングを図 3 に示す。FPGA 搭載開発ボードは、映像信号から垂直同期信号を取得する。

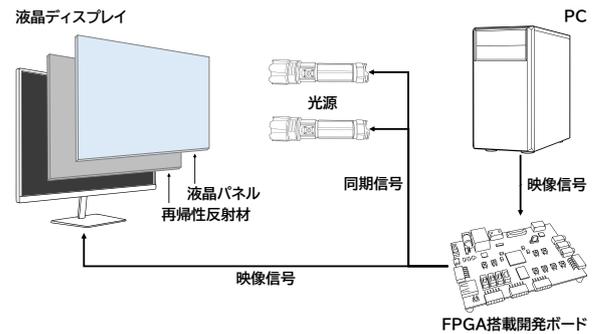


図 2 システム構成

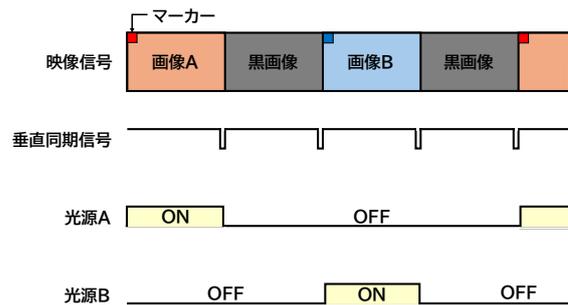


図 3 信号タイミング図

フレーム検出や映像の識別、光源の制御は垂直同期信号の立ち上がりエッジをトリガーとする。これにより、光源の点灯を、対応する映像を表示するフレームと同期することができる。

2.2 プロトタイプの実装

開発したプロトタイプについて説明する。本プロトタイプでは、提示映像として 2 種類の静止画を用いた。

液晶ディスプレイには、KTCH25X7 (24.5 インチ、解像度 1920×1080、リフレッシュレート 360 Hz) を使用した。ディスプレイのバックライトを除去し、再帰性反射材として 3M 社製スコッチライト反射布 8910 ブライト・シルバーを液晶パネル背面に配置した。FPGA 搭載開発ボードには、Digilent 社製 Zybo Z7-10 を使用した。光源には LED 懐中電灯 (Shadowhawk S1476) を採用し、内部の駆動回路に搭載されている MOSFET のゲート端子に対して、FPGA 搭載開発ボードの GPIO から出力される光源同期信号が入力されるよう改造を施した。PC には、NVIDIA GeForce RTX 5090 を搭載した Windows デスクトップ PC を使用した。

PC 上で動作する描画プログラムは、Processing 4.3 を用いて開発した。本プログラムは、画像 A および画像 B の 2 種類の静止画を、1 フレームごとに交互に出力する。この際、各フレームに対応する提示映像を識別可能とするため、第 1 ピクセルの RGB 値を、画像 A では (255, 0, 0)、画像 B では (0, 0, 255) に書き換えてから出力する。

表示フレームレートを 240 fps で安定させるため、プログラムにおける目標フレームレート (`frameRate()` による設定値) を 600 fps に設定し、GPU 側のリフレッシュレート上限および Windows のディスプレイ設定におけるリフレッシュレートを 240 Hz に設定した。

FPGA 搭載開発ボードは、HDMI で映像信号の入出力を行い、解像度を 640×480、リフレッシュレートを 240Hz に設定することで帯域制限を満たした。

3. 検証

研究室所属の大学生 5 名に対して、ディスプレイ本体のフレームレートを 30 fps, 60 fps, 120 fps, 240 fps の 4 パターンに設定した場合の映像提示の比較を行った。

観察者全員が、フレームレートが高いほど点滅は速くなり、目の負担や不快感が低減したと答えた。特に 240 fps の場合、点滅を感じられない程度に高速であると全員が答えた。一方で、240 fps での提示において、不規則に点滅感の強調、映像の混濁が発生することがあった。これはフレームドロップによって同じ画像が連続して出力されることで、画像 A → 黒画像 → 画像 B → 黒画像 → …というフレーム構造が崩れることに起因すると考えられる。

4. おわりに

本研究では、Magic Light Display について、FPGA を用いて映像信号から同期信号を生成することで、高速かつ安定な光源制御を可能とする実装方法を提案した。これにより、光源の点滅を観察者が感じられない程度に高速化することができた。また、同期ずれによって光源と映像の対応関係が崩れることがなくなった。

プロトタイプでの検証によって、フレームレートが高いほど、観察時の目の負担や不快感が低減することがわかった。しかし、現在の実装方法でも、フレームドロップによる瞬間的な映像の混濁の発生は解消できていない。

このことへの対処法として、黒画像の挿入や映像の切り替えを PC ではなく FPGA で行う手法が考えられる。例えば、PC が異なる映像を 120 fps で交互に出力し、それに対し FPGA が、1 フレームごとに黒画像を挿入した 240 fps の映像をディスプレイへ出力するという方法である。この手法では、フレームドロップによってフレーム構造が崩れた場合でも、確実に黒画像を挿入することができる。また、FPGA への映像出力を 2 系統に増やし、2 台の PC からそれぞれ異なる映像を 60 fps で出力するという方法も考えられる。FPGA は、2 種類の映像と黒画像を用いて 240 fps の映像を出力することになる。この場合、PC が行うのは 60 fps での映像出力のみになり、負荷軽減によるフレームドロップの抑制や、フレーム構造の維持が期待できる。

今後は、映像の完全な分離や、高解像度での映像提示を目指す。また、Magic Light Display を応用したコンテンツの

考案を行っていく予定である。

謝辞 本研究は JSPS 科研費 24K15253 の助成を受けたものです。

参考文献

- [1] 橋本直. Magic Light Display: 再帰性反射と点滅光源による映像の隠蔽と多重化. インタラクション 2023 論文集, 2023. p. 337-339.