

令和 98 年度
プロジェクトデザイン III
プロジェクトレポート
情報工学科

WebAssembly モジュールとして
WebGPU を用いることによる
Web 上のグラフィクスレンダリングの高
速化

提出日 令和 99 年 99 月 99 日

指導教員：鷹合 大輔 准教授

氏名 学籍番号 (クラス - 名列番号)

天羽 大樹 1101081 (4EP4-04)

概要

この研究の目的は、Web アプリケーションの処理の高速化の手段として、近年注目を集めている WebAssembly (以後、WASM と記載する) という Web アプリケーション上で動作するバイナリフォーマットプログラム (またはそのプログラムを保存したファイル) を用いて、Web グラフィクスレンダリングを高速化可能かどうか、実際にどれほどの高速化が可能なのかを検証することである。

背景として、Web ブラウザ上に組み込みで搭載されている API を用いて、多様な処理を行えるようになった近年において、Web アプリケーションでも、VR,AR を用いた没入型のアプリケーションが開発できるようになった。その際、なるべくリアルタイムな没入体験をユーザへ提供するにあたって、画面上にグラフィクスをレンダリングする処理の速度が重要になると考えた、これまでそれらのようなコンテンツを作成するにあたって、開発を円滑に行うためのライブラリ等は開発され、提供されてきた。しかし、それらのライブラリよりも高速にどうさせる方法として、WASM を通して Web グラフィクスを扱うことで、高速化を図ることとした。

手法として、これまで、JavaScript（以後、JS と記載）のプログラム言語で記述され、実行可能であったプログラムを、Rust というプログラミング言語を用いて、記述しなおした上で、WASM として利用できるようにコンパイルし、WASM を JS 側からロードして、WASM でコンパイルされた関数をコールさせるという手法である。

この手法が有効であると考えられる理由として、WASM を有効活用できるとされる処理は、多くの数値演算を用いる処理であることとされている。グラフィクスレンダリングにおいて、動きをつけるようなものは、内部的に大量の行列演算を行うこととなる。そのような演算タスクをもつ処理であるならば、WASM の高速な数値演算処理の恩恵を受けることが可能であると考えたためである。

[illegible]

活動履歴 山田 太郎

| 期間 | 活動内容 | 活動時間 [h] |
|--------|------------------|----------|
| 4-8 月 | 調査・実験・実装 | 200 |
| 11-3 月 | 実験・実装・論文執筆 (3 章) | 160 |

活動履歴 鬼 太郎

| 期間 | 活動内容 | 活動時間 [h] |
|--------|------------------|----------|
| 4-8 月 | 調査・実験・実装 | 200 |
| 11-3 月 | 実験・実装・論文執筆 (3 章) | 160 |

活動履歴 目玉野 親父

| 期間 | 活動内容 | 活動時間 [h] |
|--------|---------------------------|----------|
| 4-8 月 | 調査・実験・実装 | 200 |
| 11-3 月 | 実験・実装・論文執筆 (3 章, 2.5.1 節) | 160 |

目次

| | | |
|-------|-----------------------|----|
| 第 1 章 | 序論 | 1 |
| 第 2 章 | ウェーブレット変換 | 3 |
| 2.1 | ウェーブレット | 3 |
| 2.2 | 連続ウェーブレット変換 | 3 |
| 2.2.1 | 桃太郎伝説 | 3 |
| 2.2.2 | 順変換 | 5 |
| 2.2.3 | 逆変換 | 5 |
| 2.3 | だだだだ | 6 |
| 2.3.1 | あああ | 6 |
| 2.4 | だああああ | 6 |
| 2.4.1 | 基本的な考え方 | 6 |
| 2.5 | 算術符号化 | 6 |
| 2.5.1 | 平均符号長 | 6 |
| 第 3 章 | システム構成 | 9 |
| 3.1 | だああああ | 9 |
| 第 4 章 | 実験 | 11 |
| 第 5 章 | 結論 | 13 |
| 参考文献 | | 15 |
| 付録 A | 開発したプログラム | 17 |
| A.1 | セットアップ方法 | 17 |
| A.2 | 使い方 | 17 |
| A.3 | ソースコード | 17 |
| 付録 B | iiiiiii | 19 |

図目次

| | | |
|--------|---------------------------|---|
| 図 2.1: | ツースケール関係 | 5 |
| 図 2.2: | ツースケール関係 aassad | 7 |
| 図 2.3: | トラがでた | 7 |

表目次

| | | |
|--------|--------------------------|---|
| 表 2.1: | ここに表のタイトルを書きます | 4 |
| 表 2.2: | 言語別の特色 | 6 |
| 表 2.4: | aaa | 8 |

ソースリスト目次

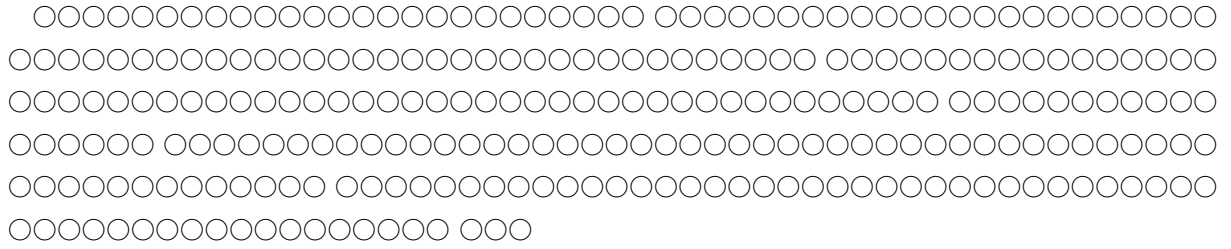
| | | |
|----------|---------------------|----|
| リスト A.1: | サンプルプログラム | 17 |
| リスト A.2: | スパゲッティソース | 17 |

[illegible]

- やまにのぼる
- せんたくにいく
- ねる

[illegible][illegible]

2.2.2 順変換



2.2.3 逆変換

まず「算術符号」の基本的な考え方について説明する．算術符号は記号列，もしくは文字列全体をひとつの符号語にする方法であり，1960 年代に P.Elias によって提案された．

算術符号は，記号列を実数 0 と 1 の間の区間を用いて表す．たとえば，記号は a,b,c の 3 種類があり，出現確率をそれぞれ $P(a) = 0.2$, $P(b) = 0.6$, $P(c) = 0.2$ とする．算術符号は，区間を記号の出現確率に比例した

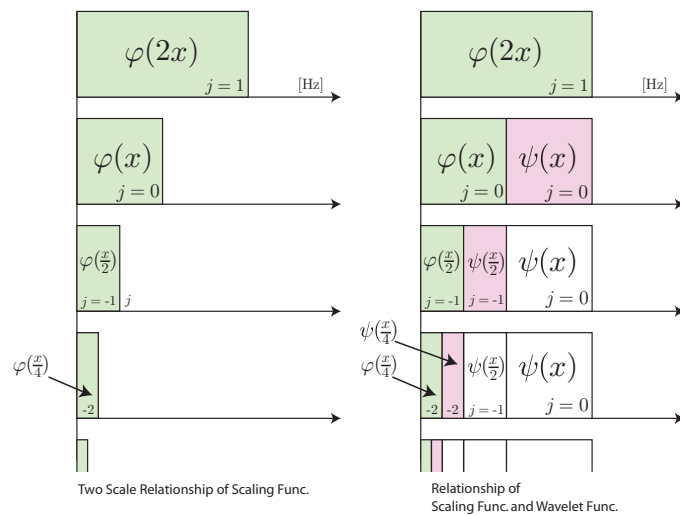
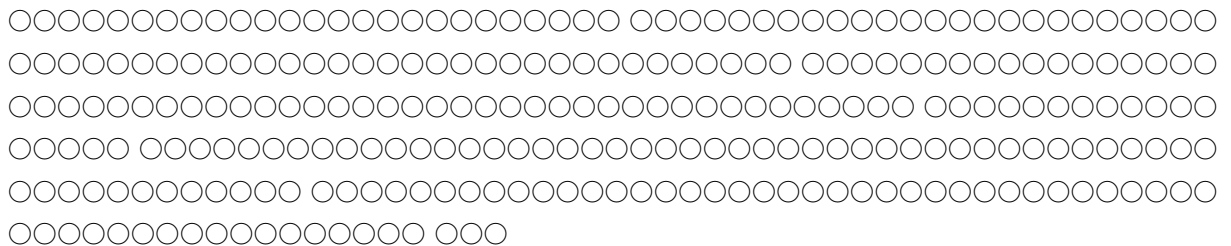


図 2.1: ツースケール関係

小区間に分割して行くことで符号化を行う．それでは例として，記号列 abbbc を符号化してみる．式 (2.1) をみてね。

$$S = \sum_{n=0}^N n^2 \quad (2.1)$$



表??に〇〇〇を示す。リスト A.2 に〇〇〇を示す。

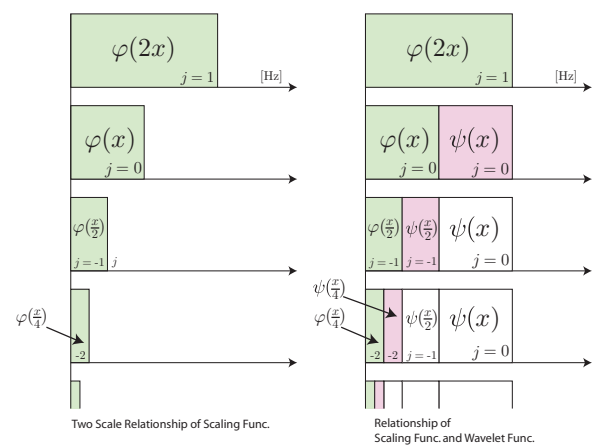


図 2.2: ツースケール関係 aassad

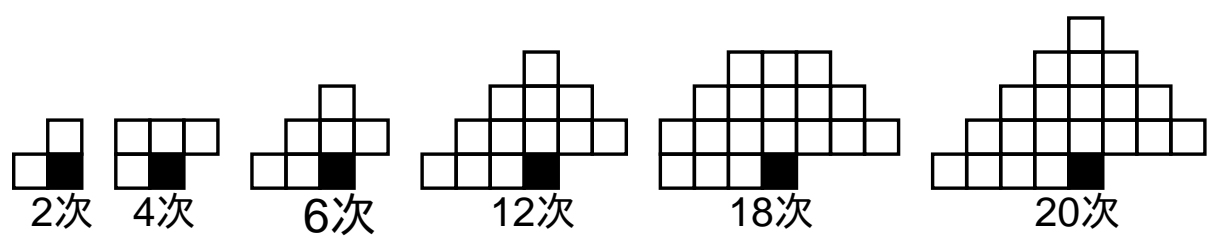


図 2.3: トラがでた

ベクトル x とただの x_0

$1 + 1 \quad \dots \text{うわあー！} \quad (2.2)$

あ

表 2.4: aaa

| | |
|---|---|
| あ | い |
| え | お |

第 3 章

システム構成

あわれといふも，なかなか疎かなり．されば，人間の働き事は，老少不定のさかいなれば，誰の人も早く後生の一大事を心にかけて，阿弥陀仏を深く頼み参らせて，念仏申すべきものなり．あなかしこ，あなかしこ．

3.1 だああああ

bababbaba

第4章

実験

あわれといふも，なかなか疎かなり．されば，人間の働き事は，老少不定のさかいなれば，誰の人も早く後生の一大事を心にかけて，阿弥陀仏を深く頼み参らせて，念仏申すべきものなり．あなかしこ，あなかしこ．あ
あわれといふも，なかなか疎かなり．されば，人間の働き事は，老少不定のさかいなれば，誰の人も早く後生の一大事を心にかけて，阿弥陀仏を深く頼み参らせて，念仏申すべきものなり．あなかしこ，あなかしこ．わ
れといふも，なかなか疎かなり．されば，人間の働き事は，老少不定のさかいなれば，誰の人も早く後生の一大事を心にかけて，阿弥陀仏を深く頼み参らせて，念仏申すべきものなり．あなかしこ，あなかしこ．

あわれといふも、なかなか疎かなり。されば、人間の働き事は、老少不定のさかいなれば、誰の人も早く後生の一大事を心にかけて、阿弥陀仏を深く頼み参らせて、念仏申すべきものなり。あなかしこ、あなかしこ。あわれといふも、なかなか疎かなり。されば、人間の働き事は、老少不定のさかいなれば、誰の人も早く後生の一大事を心にかけて、阿弥陀仏を深く頼み参らせて、念仏申すべきものなり。あなかしこ、あなかしこ。あわれといふも、なかなか疎かなり。されば、人間の働き事は、老少不定のさかいなれば、誰の人も早く後生の一大事を心にかけて、阿弥陀仏を深く頼み参らせて、念仏申すべきものなり。あなかしこ、あなかしこ。あわれといふも、なかなか疎かなり。されば、人間の働き事は、老少不定のさかいなれば、誰の人も早く後生の一大事を心にかけて、阿弥陀仏を深く頼み参らせて、念仏申すべきものなり。あなかしこ、あなかしこ。あわれといふも、なかなか疎かなり。されば、人間の働き事は、老少不定のさかいなれば、誰の人も早く後生の一大事を心にかけて、阿弥陀仏を深く頼み参らせて、念仏申すべきものなり。あなかしこ、あなかしこ。

参考文献

- [1] V.D. Vaughen and T.S. Wilkinson, "System considerations for multispectral image compression designs," IEEE Signal Process. Mag., vol.12, no.1, pp. 19-31, Jan. 1995.
- [2] A. Said and W. Pearlman, "An image multiresolution representation for lossless and lossy compression," IEEE Trans. Image Process., vol.5, no.9, pp.1303-1310, Sept. 1996.
- [3] 小野文孝, "静止画符号化の新国際標準方式 (JPEG2000) の概要," 映像情報メディア学会誌, vol.54, no.2, pp.164-171, Feb. 2000.
- [4] D. Tretter and C.A. Bouman, "Optimum transform for multispectral and multilayer image coding," IEEE Trans. Image Process., vol.4, no.3, pp.296-308, March 1995.
- [5] F. Amato, C. Galdi and G. Poggi, "Embedded zerotree wavelet coding on multispectral images," IEEE Proc. ICIP 97, vol.1, pp.612-615, 1997.
- [6] B.R. Epstein, R. Hingorani, J.M. Shapiro and M. Czigler, "Multispectral KLT-wavelet data compression for Landsat thematic mapper images," Proc. Data Compression Conf., IEEE Computer Society Press, pp.200-205, 1992.
- [7] J.M. Shapiro, S.A. Martucci, and M. Czigler, "Comparison of multispectral Landsat imagery using the embedded zerotree wavelet (EZW) algorithm," DLPO at Image Compress. Appli. & Innovation Workshop, pp.105-113, March 1994.
- [8] J.A. Saghi, A.G. Tescher, and J.T. Reagan, "Practical transform coding of multispectral imagery," IEEE Signal Process. Mag., vol.12, no.1, pp.32-43, Jan. 1995.
- [9] J. Lee, "Optimized quadtree for Karhunen-Loeve transform in multispectral image coding," IEEE Trans. Image Process., vol.8, no.4, pp.453-461, April 1999.
- [10] B. Brower, B. Gandhi, D. Couwenhoven and C. Smith, "ADPCM for advanced LANDSAT downlink applications," in Proc. 27th Asilomar Conf. Signals, Systems and Computers, Nov. 1993.
- [11] N.D. Memom, K. Sayood and S.S. Magliveras, "Lossless compression of multispectral image data," IEEE Trans. Geosci. and Remote Sensing., vol.32, no.2, pp.282-289, March 1994.
- [12] S. Gupta and A. Gersho, "Feature predictive vector quantization of multispectral images," IEEE Trans. Geosci. and Remote Sensing., vol.30, no.3, pp.491-501, May 1992.
- [13] K. Irie and R. Kishimoto, "A study on perfect reconstructive subband coding," IEEE Trans. Circuits Syst. Video Technol., vol.1, no.1, pp.42-48, March 1991.
- [14] 小松 邦紀, 瀬崎 薫, 安田 靖彦, "濃淡画像の可逆的なサブバンド符号化法", 信学論 (D-II), vol.J78-D-II, no.3, pp.429-436, March 1995.
- [15] A.R. Calderbank, I. Daubechies, W. Sweldens and B. Yeo, "Lossless image compression using integer to integer wavelet transforms," IEEE Proc. ICIP 97, vol.1, pp.596-599, 1997.
- [16] F.A.M.L. Bruekers and A.W.M.V.D. Eenden, "New networks for perfect inversion and perfect reconstruction," IEEE J. Sel. Areas Commun., vol.10, no.1, pp.130-137, Jan. 1992.
- [17] 小松 邦紀, 瀬崎 薫, "可逆的離散コサイン変換とその画像情報圧縮への応用," 信学技報, vol.IE97-83, pp.1-6, Nov. 1997.
- [18] K. Komatsu and K. Sezaki, "Design for lossless block transforms and filter banks for image coding," IEICE Trans. Fundamentals, vol.E82-A, no.8, pp.1656-1664, Aug. 2000.
- [19] 福岡 慎治, 岩橋 雅宏, 神林 紀嘉, "可逆的色変換を用いた色彩画像の可逆符号化," 信学論 (D-II),

- vol.J81-D-II, no.11, pp.2680-2684, Nov. 1998.
- [20] T. Nakachi, T. Fujii, and J. Suzuki, "A unified coding algorithm of lossless and near-lossless color image compression," IEICE Trans. Fundamentals, vol.E83-A, no.2, pp. 301-310, Feb. 2000.
- [21] 仲地 孝之, 藤井 竜也, "可逆 KL 変換を用いた病理顕微鏡画像符号化法の研究," 2000 信学総大, D16-13, March 2000.
- [22] 鷹合 大輔, 武部 幹, "可逆 WT・KLT を用いるマルチスペクトル画像の情報圧縮," 信学論 (A), vol.J84-A, no.3, pp.1-11, March 2001.
- [23] 尾上 守夫, "画像処理ハンドブック," 昭晃堂, pp.554-561, 1987.
- [24] 鷹合 大輔, 武部 幹, "マルチスペクトル画像のバンド間・バンド内相関除去による可逆情報圧縮," 第 15 回 DSP シンポジウム講演論文集, pp.475-48, Nov. 2000.
- [25] 鷹合 大輔, 武部 幹, "ウェーブレット変換を用いたマルチスペクトル画像の情報圧縮符号化法の研究," 平成 10 年度金沢工大卒業論文, 1998.
- [26] 酒井 幸市, "デジタル画像処理入門," コロナ社, 1997.
- [27] 榊原 進, "ウェーブレットビギナーズガイド," 電機大出版局, 1995.
- [28] 武部 幹, "情報圧縮、通信と回路理論," 信学技報, IT97-40, July 1997.
- [29] Gilbert Strang and Troung Nguyen, "Wavelet and filter banks," Wellesly Cambridge Press, 1996.
- [30] 貴家 仁志, "よくわかるデジタル画像処理," CQ 出版社, 1996.
- [31] "金沢の暮らし", <http://www.kanazawa-it.ac.jp>
- [32] 山田 太郎, "金沢の一人暮らし", トンチンカン出版, 2016.

付録 A

開発したプログラム

A.1 セットアップ方法

にプログラムの使い方, セットアップ方法などを書きましょう. ここにプログラムの使い方, セットアップ方法などを書きましょう. ムの使い方, セットアップ方法などを書きましょう. ここにプログラムの使い方, セットアップ方法などを書きましょう.

```
$ sudo apt-get install python3-pip      # PIP コマンドの導入
$ echo "Hello WOrld"
Hello WOrld
```

にプログラムの使い方, セットアップ方法などを書きましょう. ここにプログラムの使い方, セットアップ方法などを書きましょう. ムの使い方, セットアップ方法などを書きましょう. ここにプログラムの使い方, セットアップ方法などを書きましょう.

A.2 使い方

ここにプログラムの使い方, セットアップ方法などを書きましょう. ここにプログラムの使い方, セットアップ方法などを書きましょう. ここにプログラムの使い方, セットアップ方法などを書きましょう. ムの使い方, セットアップ方法などを書きましょう. ここにプログラムの使い方, セットアップ方法などを書きましょう. ここにプログラムの使い方, セットアップ方法などを書きましょう.

A.3 ソースコード

リスト A.1: サンプルプログラム

```
1 #include<stdio.h>
2 int main(){
3     return 0; // ウギャー！
4 }
```

リスト A.2: スパゲッティソース

```
1 #! /usr/local/bin/ruby -Ks
2 # numbers.rb
3 print "正の整数値を表す文字列を入力してください。正の整数値を表す文字列を入力してください。 \n"
4 while true
5     print ">"
6     line = gets.chomp # 改行コードを切り捨てる
7     break if line.empty?
8     begin
9         v = Integer(line) # 文字列を整数化する
10    rescue
11        puts "変換できません。"
12        next
13    end
14    printf ("2進法: %b\n", v)
```

```
15   printf ("8進法：%o\n",v)
16   printf ("10進法：%d\n",v)
17   printf ("16進法：%x\n",v)
18 end
19 puts "Bye."
```

付録 B

しいしいしい

ああああああああああああああああいはいいはいはいはいはいはいはいはいはいはいはいはい
いういういういういういういういういう