

目次

1	緒言	1
2	e-Learning	2
2.1	e-Learning の定義	2
2.2	e-Learning に含まれる学習形態	2
2.2.1	CAI	2
2.2.2	CBT(Computer-Based Training)	3
2.2.3	WBT	3
2.2.4	EPSS(Electronic Performance Support System)	3
2.2.5	Knowledge Management System	4
2.2.6	Blended Learning	4
2.3	e-Learning の効果	4
2.3.1	e-Learning のメリット	4
2.3.2	e-Learning のデメリット	5
3	シミュレータ教材	7
3.1	シミュレータ教材の定義	7
3.2	FDTD 法	7
3.2.1	FDTD 法の概要	7
3.2.2	計算方法	7
3.3	シミュレータ教材のメリット	9
3.4	シミュレータ教材のデメリット	9
4	プロセッサ	11
4.1	CPU	11
4.1.1	CPU の概要	11
4.1.2	CPU の歴史	12
4.1.3	CPU の処理方法	12
4.2	GPU	13
4.2.1	GPU の概要	13
4.2.2	GPU の歴史	13
4.2.3	GPU の処理方法	14
4.2.4	GPU の活躍	14
4.3	CPU と GPU の相違点	14
5	プログラム	15

5.1	JavaScript	15
5.2	WebWorker	15
5.3	OffscreenCanvas	15
6	実装したシミュレータ	16
6.1	開発したシミュレータ	16
6.2	実験方法	16
7	実験結果	17
8	結言	18
9	謝辞	19
10	参考文献	20
11	付録 制作したプログラム	21

図目次

1	CPU の内部構造の略図	11
2	グラフィックボード	14

表目次

1 緒言

近年 e-Learning の普及とともに様々なシミュレータ教材の需要が増加している。さらに、パソコンだけでなく、タブレットやスマートフォンといった様々な端末での利用が見込まれ、その利用は学校だけでなく、塾や家庭など幅広い場での活躍が考えられる。

しかし、スマートフォンの性能は PC と比較して決して処理速度が速いとは言えず、一部の塾や通信講座では独自のハードウェアやアプリケーションの開発を行っていることから、従来のシミュレータ教材をそのまま適用するのは容易ではないのが現状である。したがって、従来のシミュレータ教材の処理速度を改善する必要がある。しかし、全てのシミュレータ教材を新規で開発した場合その労力は計り知れない。そのため、ただ処理速度を向上させるだけでなく、いかに変更点を少なく、処理速度を上げるのかが重要となってくる。現在処理速度を向上させる手法として主にシェーダや Worker を用いる方法がある。

しかし、シェーダと Worker は共に問題点がある。シェーダは処理速度が大幅に向上するが、記述内容が複雑、増加するため手軽に実装することができないといった問題点がある。Worker は JS をマルチスレッド化させる API である。しかし、描画処理を行うことができないため worker で描画に必要なデータを計算で解き、別のファイルで描画を行う必要がある。そのため従来のソースコードを処理と描画で分ける必要がある。それらのデータの送受信を実装する必要があり、決して手軽に実装できないという問題点がある。

本研究室では主に音響のシミュレータ教材を制作し配布を行っていることを受け、音響のシミュレータで広く普及している FDTD 法を用いたシミュレータ教材を利用する。

本研究では FDTD 法により制作されたシミュレータ教材を手軽に処理速度を向上させる手法を考じ、制作、有用性を示すことを目的とする。

2 e-Learning

e-Learning とは electronic Learning の略称である。その名の通りコンピュータやネットワークなどの情報技術を扱う学習形態である。e-Learning は 1999 年 11 月、アメリカフロリダ州にて開催された TechLearn1999 において初めて使用された。現在 CAI やオンラインラーニングなど様々な学習形態が存在する。

2.1 e-Learning の定義

米国の組織学習・人材開発に関する世界最大の会員制組織である ASTD(American Society for Training & Development) によると「e-Learning とは、明確な学習目的のために、エレクトロニクス技術によって提供、可能とされた伝達されるあらゆるものである。」と、定義されている。

一般的に情報技術を用いているもの全てを e-Learning と「広義」でとらえる場合と、非同期型オンライン方式を想定した「狭義」である場合とさまざまであり、一概に定義通りではない。テクノロジーを活用した学習形態の幅が広がる現在の傾向では「情報技術によるコミュニケーション・ネットワークなどを活用した主体的な学習」を総称して e-Learning と定義するのが一般的である。

2.2 e-Learning に含まれる学習形態

e-Learning は IT 技術を用いた様々な学習形態を総称して呼ぶ。したがって一言で e-Learning と呼んでも複数の種類が存在する。そこで本章では e-Learning の種類と特徴・開発の経緯などを紹介する。

2.2.1 CAI

CAI(Computer-Assisted Instruction) は 1950 年代後半、米国で兵員教育の目的として発足。1970 年代から 80 年代にかけて、「講師と受講者が長時間同じ場所に居なければならないという問題を、コンピュータを用いることにより軽減できないだろうか」という、パソコンを教育に活用する学習形態として注目された。CAI はスタンドアローン環境が前提であり、受講者はマニュアル通りの単純な、決められた学習を行う形式だった。そのため受講者のレベルに応じた教育、柔軟なカリキュラムの提示を行えず、効率的な学習を十分に行えなかった。また、ネットワークの利用には至らなかったため、受講者の進捗の確認などは人間が行う必要があった。

1980 年代に入るとハードウェアの開発が進み、従来の大型コンピュータで実行されていた CAI がパーソナルコンピュータでも実行可能となり、モデムやスキャナーなどの周辺機器の開発によりメディアとしての能力が向上した。

1990 年代にはコンピュータネットワークが発達、個々人の情報リテラシー能力の向上を受け、CBT や WTB が考案、2000 年代以降には、これらの概念を活用した e-Learning が盛んに用いられるようになった。

2.2.2 CBT(Computer-Based Training)

CBT(Computer-Based Training) は、1986 年に教育、調査、測定分野の非営利団体である ETS(Educational Testing Service) が、大学生の能力別クラスの編成用のテストとして利用したことから始まった。当時はネットワークが普及していなかったため、CD-ROM の大容量な特性を活かし、動画や音声などを利用したインタラクティブなコンテンツが効果的に利用された。また、受講者管理やコンテンツ配信を行うサーバを必要としないため、比較的容易に導入することが可能だった。しかし、CD-ROM を制作するためのコストと、配布後の修正が難しく、各個人の進捗状況を一括して管理することが困難であった。

その後、1990 年代に入り、IT 系企業の認定資格試験に利用されたことから大きく発展し、現在では公的要素の強い試験など様々な分野で CBT 化が進んでいる。日本国内においても国家試験の一つである情報処理技術者試験で 2003 年から導入されているように、現在でも学校教育や企業内教育の現場で、幅広く取り入れられている。

2.2.3 WBT

1990 年代中盤から後半にかけて、ソフトウェアの進化とネットワークの普及に伴い、インターネットやイントラネットなどのネットワークを通して教育コンテンツ学習者に提供する WBT(Web-Based Training) という学習形式である。従来の WBT と CBT との大きな違いは、ネットワークを介していることである。ネットワークを用いることによる恩恵は以下の 3 つが挙げられる。

1. 知識の変化に合わせて教育内容の更新が容易なこと
2. 双方向通信が可能になり、受講者と講師、もしくは受講者同士間のインタラクティブなコミュニケーション、及び受講者の進捗状況の把握が可能になったこと
3. Web は世界標準のプロトコルが存在するため、教材互換性、汎用性、操作性が向上したこと

2.2.4 EPSS(Electronic Performance Support System)

EPSS(Electronic Performance Support System) とは、IT 技術を活用して、業務中に必要な知識やツールの提供を行うことで、パフォーマンスの向上を支援するシステムのことである。

例としてコールセンターやビデオマニュアルが挙げられる。インタラクティブ学習やパフォーマンス・サポート・システムの分野を専門にしているグロリア・ゲーリー氏によれば「他社による最小限のサポートと介入で、ジュブパフォーマンスが生まれることを可能にするモニタリングシステム、情報、ソフトウェア、支援、データ、画像、ツール、評価の全ての内容に働く人々が簡単にアクセスできる。また、それらを個々人が手元で見ることができ、直ちに必要な内容を入手できる仕組みを持つ、使いやすく統合された電子的環境」と定義している。

CBT や WBT といった、業務から一度離れて学習する形態と比較し、より業務遂行に直結した知識を会得できるシステムであることが特徴であり、近年はより効率化を図るために、

これまで活用していた紙のマニュアルを電子化する動きがある。米国においてはユーザのパフォーマンスに合わせた設計を行うことを PCD(Performance Centered Design) と呼び、EPSS と同義として扱われる場合がある。

2.2.5 Knowledge Management System

Knowledge Management とは、個人の持っている知識、情報を組織で共有し、コミュニティ全体の創造性を向上させるための手法である。個人の持つデータだけでなく、経験やノウハウを含んだ幅広い物を指す。

Knowledge Management と e-Learning は独立したものと扱われることが多かった。しかし、米国で e-Learning の分野においても、マニュアルの理解や受講者間での知識の共有や生成の場を設けることの重要性を訴えられる様になった。そのことが実現した場合 e-Learning と Knowledge Management の区別がなくなるため、両者を融合し、e-Learning の一環として捉えるようになった。

2.2.6 Blended Learning

Blended Learning は、e-Learning と集合研修を組み合わせた形態ことである。

それらの組み合わせは、他者からの刺激を受けることで学習意欲の向上、相互作用をによって理解促進、知識の整理が行えるという利点がある。

e-Learning + 集合研修

事前学習を済ませた後に、教室でインタラクティブな学習を行う

e-Learning + 集合研修

事前学習と教室研修を行い、その後にバーチャルクラスで学習を行う

集合研修 + e-Learning

集合研修後にフォローアップのためにセルフ学習を行う

ディスカッションバーチャルクラス、チュートからのメンタリングを含む

このように様々な形態が存在するが、e-Learning に適した内容化を見極める必要である。

2.3 e-Learning の効果

e-Learning には様々な形態が存在するが、学習形態ごとに違いがあるが、e-Learning という大きな枠組みで考えたときのメリットとデメリットをまとめる。

2.3.1 e-Learning のメリット

受講者のメリット

1. 場所と時間に制限がない
2. 受講者のレベルや習熟度に合わせたコンテンツの提供できる
3. 研修終了後に受講者間や講師との間でインタラクティブなコミュニケーションが可能
4. 世界標準のプロトコルを用いることで、教材の互換性、汎用性、及び操作性が向上

5. 予め筋道を立ててシステムを組むため、全体の筋道が決まっており、一括して見やすい
6. 場所の移動を省くことが可能なため、時間とコストを削減できる
7. 最新の情報を安価で早く入手し、学習することができる

制作側のメリット

1. 集団教育と比較すると安価に提供できる
2. マルチメディアを用いたインタラクティブな教材の制作が可能
3. 学習の進捗状況をリアルタイムで把握することができるため、管理が用意
4. 知識の変化に合わせ、最新の内容に更新した教材の把握が容易

2.3.2 e-Learning のデメリット

- (1) コンピュータ、またはインターネット環境が整っているところでしか学習できない

本などの資料は、持ち運びが容易なため、電車の中などで資料を持ち込むことや書き込むことで学習することが可能だが、e-Learning では難しい。しかし、小型の端末などで学習を行えるソフトがリリースされており、以前と比べるとその欠点は薄れつつあるといえる。

- (2) 就業時間中の学習を公認する必要がある

集合研修は、会社がその時間の学習を公認しているが、書籍での学習は就業時間外として扱われることが多く、自主学習形態をとることが多い。しかし、e-Learning はインターネットが使えるコンピュータで学習することが一般的であり、セキュリティ上社外からのアクセスは禁じられていることが多く、学習するには社内で行う必要がある。その場合就業中か、仕事を終えて自分の席で学習することになる。自分の席で学習する場合、他者の目にはばかられ、思うような学習は難しいと思われる。就業時間中の学習は多くの企業で公認されにくいいため、教材をリリースしても誰も学習しないという状況も多い。

- (3) システム投資コストがかかる

e-Learning を使用する環境を構築するには教材を制作するだけでなく、サーバ管理コストやコンテンツ維持コストや補助的サーバなどが挙げられる。また、レポート提出や質疑応答システムと統合した環境を整えるとそれなりに費用がかかってしまいます。システムの投資コストが従来の教育コストより高くなってしまう場合、e-Learning のメリットが無くなってしまう。

- (4) コミュニケーションや、体で覚える技術教育には向かない

実際に対話を行う集合研修に対し、e-Learning は成約が多い。しかし、誰かが一方的に話すタイプの集合研修と比較すれば、この弱点は解消されたといえる。

- (5) シミュレータ搭載が困難

現在実用化されている e-Learning 教材のなかで、文系科目に該当する大学などで多く実用化されている。その背景に容易に制作が可能なが挙げられる。容易に作成

できることからコンバータを用いることが多いが、コンバータはシミュレータやアニメーションに対応されておらず、実験などの補修学習が要の理系科目において十分な効果を期待することはできない。

3 シミュレータ教材

3.1 シミュレータ教材の定義

シミュレータ教材とはシミュレーションを行う教材のことである。

現実世界の問題はシステムが複雑で予測が困難な場合が多く、実システムで試すにはコストや時間がかかるといった場面が多く存在する。しかし、コンピュータ上に現実世界を再現することにより、実際のシステムの制作、変更することなく様々な条件下の環境を再現、システムの挙動を調べることができる。このコンピュータ上に現実世界を仮想的に制作するシステムをシミュレータと呼ぶ。

したがって、シミュレータ教材とは様々な条件下の現象を実際に生成することなく再現することで、対象を視認、理解させるための教材である。

一般的には、航空シミュレータや音響シミュレータなど、シミュレーションの結果を体験させるためのシステムを指すことが多いが、本研究では、コンピュータ上で動作し、ブラウザ上でパラメータを変更させ、その結果を同じブラウザ上で確認できるソフトウェアのことを指す。

3.2 FDTD 法

音場のシミュレーションは大きく分けると定常解析手法と非定常解析手法の2つに分類される。定常解析手法とは対象が定常状態に達したときの音場や位相を計算する手法である。音波は時間とともに変動するが定常状態のため時間に依存することなく空間分布が求められる。

一方、非定常解析手法は時間によって変化する場を取り扱う。したがって音圧などの物理量が時間と共に変化する場を扱うため、波形そのものの計算が得意である。

本章では本研究に使用し、音響のシミュレータにおいて度々用いられる非定常手法の一つである FDTD 方について論じていく。

3.2.1 FDTD 法の概要

FTDT(Finite Difference Time Domain) 法は、1966 年に K.S.Yee によって提案された。K.S.Yee はマクスウェル方程式が電解と磁界の連立方程式であることに着目し、中心差分に電磁界の時間及び空間配置を考案した。後に多くの科学者によって発展し、現在では電磁波の支配式を求めるだけでなく、流体力学や熱工学だけでなく波動光学にも用いられることが多くなった。

3.2.2 計算方法

物体に力を加えると物体は移動する、それと同様に空気も力が加わると動きが生じる。それはニュートンの第2方式、すなわち、運動方程式で記述される。空気には物体に動きが加わることによって生じる圧力と、運動に依存しない圧力が存在し、音響において前者を音圧、後者を大気圧と呼ぶ。音波の伝搬する空間を音場と呼ぶ。

ある空間において x, y の直交座標の寸法がそれぞれ $\Delta x, \Delta y$ とし、密度を ρ 、音圧を p とし、それぞれの変異を u_x, u_y とする。 Δx が微小であると仮定したときの x 方向の運動方程式は

$$\rho \frac{\partial^2 u_x}{\partial t^2} = -\frac{\partial p}{\partial x} \quad (1)$$

y 方向も同様に

$$\rho \frac{\partial^2 u_y}{\partial t^2} = -\frac{\partial p}{\partial y} \quad (2)$$

と表される。体積弾性率を $k[N/m^2]$ としたときの音圧に関する連続方程式は

$$p = -k \left(\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} \right) \quad (3)$$

となる。式 (1), (2), (3) を物理現象を方程式化した式、すなわち、支配式として音場の解析を行う。これらの式を時間微分した式を以下に示す。

$$\rho \frac{\partial v_x}{\partial t} = -\frac{\partial p}{\partial x} \quad (4)$$

$$\rho \frac{\partial v_y}{\partial t} = -\frac{\partial p}{\partial y} \quad (5)$$

$$\frac{\partial p}{\partial t} = -k \left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} \right) \quad (6)$$

となる。

音圧は時間と共に連続的に変化するが、コンピュータは連続の関数を扱うことができない。したがって、空間や時間を細かく区切ることで、連続な関数に見せる必要がある、このことを離散化と呼ぶ。また、空間に関する区切り幅を空間離散化幅、時間に関する区切り幅は時間離散化幅と呼ぶ。区切りごとの離散的な値を参照点と呼ぶ。

FTDT 法における音圧の離散化は、隣り合う参照点がそれぞれ $\Delta x, \Delta y$ の空間離散幅を持ち、 Δt の時間離散幅を持つ。参照点から x, y 方向に関して何番目の音圧参照点かを i, j を空間ステップと呼ぶ。何番目の時間参照点であるか n を用いて表すと、 $(n - 0.5)\Delta t[s]$ であり、この n を時間ステップと呼ぶ。

x に関する偏微分係数を

$$-\frac{\partial p}{\partial x} \Big|_{x=x_0} = \lim_{\Delta x \rightarrow 0} \frac{p|_{x=x_0+\frac{\Delta x}{2}} - p|_{x=x_0-\frac{\Delta x}{2}}}{\Delta x} \sim \frac{p|_{x=x_0+\frac{\Delta x}{2}} - p|_{x=x_0-\frac{\Delta x}{2}}}{\Delta x} \quad (7)$$

のように近似する。式 7 を式 4 に適用し、空間ステップを $i+0.5, j$ の位置、空間ステップ n の時刻としたとき

$$\rho \frac{v_x^{n+0.5}(i+0.5,j) - v_x^{n-0.5}(i+0.5,j)}{\Delta t} = - \frac{p^n(i+1,j) - p^n(i,j)}{\Delta x} \quad (8)$$

と近似できる．空間微分にも時間微分にも式 7 と同様の近似を用いることが可能となり，これが FDTD 法の特徴の一つである．式 5 も同様に近似値を求めることができる．

$$\rho \frac{v_y^{n+0.5}(i+0.5,j) - v_y^{n-0.5}(i+0.5,j)}{\Delta t} = - \frac{p^n(i+1,j) - p^n(i,j)}{\Delta y} \quad (9)$$

時間ステップ $n-0.5$ を過去， n を現在， $n+0.5$ を未来の状態と考えれば，過去の v_x の値と現在の値から未来の v_x の値を求めることができる．最も時間ステップが大きい項のみ左辺に変形し，初期の粒子速度分布と音圧分布の 1 組がわかれば，全空間ステップにおいて変形した式を交互に計算することで求めることが可能である．初期時刻の場の業態を表す条件のことを初期条件と呼ぶ．時間が進むごとに次々と変化することを時間発展と呼び，初期状態から時間的な順序を追って場の上程を求めることを逐次計算と呼ぶ．

3.3 シミュレータ教材のメリット

シミュレータを使用することに得られるメリットを以下に示す．

1. コンピュータで疑似体験

シミュレータ教材とは，現実の物理現象をモデル化し，コンピュータ上で計算することで，模擬体験できる藻である．仮想環境の中での学習により，従来では考えられない程の学習効果を得ることができ，学習者の意欲向上に繋がると考えられる．

2. 等しく均一の条件で学習が可能

学習者自身が，実験の条件や変数を変えながら実験に参加することが可能である．実験変数の変更を安全に，何度でも変更することが可能なため，繰り返し学習が可能となり，学習者は全員均一の条件で学習できる．

3. 安全性

シミュレータ教材により学習者は，教室内で費用，時間のかかるものや，危険で簡単に実施できない実験を疑似体験することができる．

3.4 シミュレータ教材のデメリット

シミュレータ教材は通常，全ての現象を思考要素とせず，対象要素を絞り込むことにより，要素が現象に与える影響を検証することが目的である．したがって，結果が不確定な事象を検証することは極めて困難である．

コンピュータを用いた積算によるシミュレートは，基本線形近似による計算のため，非線形を含む自然現象をシミュレートする場合は必ず誤差が生じる．良好な結果が得たければ，誤差見積もりが重要になる．

そして、シミュレータ教材を使うにはシミュレータ教材を作る必要がある。開発者はシミュレータを開発するためのソフトウェアを保持した状態で、プログラミングの知識や主題についての専門的な知識が要求されるため、開発者が限定される。

4 プロセッサ

本章ではプロセッサ，CPU や GPU の歴史や相違点といった内容を解説する。

4.1 CPU

CPU とは Central Processing Unit の略称である。人間でいう頭脳と例えられることが多いパーツである。本章ではその概要と歴史について記す。

4.1.1 CPU の概要

CPU の外見は大きさが数 cm, 厚さ数 mm で，容器に覆われている。裏面には外部との接続のため，数多くのピンが設けられている。このピンを介して，CPU はプログラムを読み込み実行するのである。

CPU の役割はメモリやハードディスク，マウスなどからデータを受け取り，計算，判断をし，その結果をディスプレイやメモリ，ハードディスクに送り出すことである。それらの命令を実行するためには図 1 のように複数の演算ユニットが必要になる。CPU は，外部から供給されるクロック信号に合わせて内部動作を進め，その信号は数 GHz と非常に高いなっている。



図 1 CPU の内部構造の略図

4.1.2 CPU の歴史

パソコンが誕生する前のコンピュータは非常に高価なため、個人が所有できる代物ではなかった。しかし、1971年にインテルが4ビット CPU4004の開発に成功したことにより、以前より手軽にコンピュータを入手できるようになった。このCPUは電卓メーカーのビジコンが電卓の制御用に依頼したものである。1974年には更に性能が向上した8080CPUを開発した。同年MITS(マイクロ・インスツルメンテーション・テレメトリ)が本CPUを利用したコンピュータALTAIR8800を販売した。8080は汎用性が高く、様々な製品に組み込まれた。

1978年に16ビットCPUのIntel 8086が登場した。その後、80286、Intel 386、Intel 486そして、Pentiumへと進化していった。これらの特徴として上位互換性があることである。つまり、以前に作られたOSやアプリケーションを動かすことができることである。この一連のCPUはCISC(Complex Instruction Set Computer)タイプのCPUと分類される。CPUは、加法、減法命令、メモリの読み書きなど、各種命令を実行するのが目的である。実行可能な命令はCPUごとに異なり、CPUが実行可能な命令を全部合わせて命令セットと呼ぶ。CISCは複雑命令セットコンピュータと呼ばれ、それぞれの命令の機能が複雑かつ強力に作られている。1980年RISC(Reduced Instruction Set Computer)が発表された。RISCは命令を使用頻度の高いものに限りられ、基本設計がシンプルなのが特徴である。その後、パソコンにおいて従来の互換性を保持しつつ、RISC技術を取り入れたインテルが優位に立ち、ワークステーションにおいてはRISC CPUに軍配が上がった。1990年代に入ると業務用向けに64ビットのCPU MIPS R4000が登場した。家庭用のCPUは32ビットへの以降が進んでいた。

1990年代後期にはサーバ向けCPUの64ビット化が一区切りし、高クロック数、マルチプロセッシングへと変化していった。1993年にインテルがPentiumシリーズを発売、初期のクロック周波数は60MHzだった。その後200MHzまで向上する。2000年に入りAMDがAthlonを発表、最大周波数1GHzになった。同年インテルはAthlonに対抗してPentium 4を発表、動作周波数は1.3GHzだった。

2002年にサーバ分野でマルチコアCPUが導入された。翌年パソコン分野において64ビットのCPUが登場した。2004年に周波数の向上に限界に突き当たった。周波数を向上させることにより、消費電力が増加した。そこでインテルは2005年に1つのパッケージに2つのCPUコアを実装したCPUを開発、Pentium Dの登場である。このCPUがマルチコア化への先駆けとなった。同年AMDからマルチコアのAthlon 64 X2が発表された。AMDはクロック数が1GHzへ到達した時点で処理効率を重視したCPUを展開した事により、デュアルコアへの拡張を意識した設計を可能とした。この頃から消費電力あたりの性能が重視されるようになった。サーバ向けのCPUではワンチップで数十スレッドを実行するCPUが現れた。相対的に低いクロック数で性能を引き出しやすいSIMD(single instruction multiple data)の性能に重点が置かれた。

4.1.3 CPU の処理方法

MMX

コンピュータ内で処理されているフルカラー処理は、24 ビットで行われている。24 ビットを処理するには 8 ビットを 3 回行うか、8 ビットをパディングしてデータを処理する。各要素をバラバラに処理する場合 3 回演算を行うため非常に時間がかかった。

そこで 64 ビットレジスタを 32 ビットを 2 つ、16 ビットを 4 つとみなし、それらの領域をまとめて演算するという方式が用いられた。32 ビットを 2 つ処理するが、同じ処理を 2 回行うわけではなく、まとめて処理を行う。この処理を MMX と呼び、現在では SSE と呼ばれている。

パイプライン

4.2 GPU

GPU は Graphics Processing Unit の略称である。元々 GPU は画像処理、大量の計算を行うことを目的として開発されたハードウェアである。しかし、近年はその処理能力の高さからディープラーニング等画像処理以外の分野での活躍も多く見られるようになった。本章では、GPU の概要から歴史、処理の仕方、役割について述べる。

4.2.1 GPU の概要

GPU は元々画像処理を行うことを目的としている並列演算ハードウェアである。NVIDIA によってグラフィックス処理以外の演算を行わせるようにした GPGPU(General-Purpose computing on GPUs) の開発環境 CUDA が一般公開されると、GPGPU の研究が注目を浴び、現在ではその高い処理能力を活かしディープラーニングに広く普及している。

一般的に GPU とは、図 2 のようなビデオカード (グラフィックボード) と呼ばれるハードウェアとして販売されている機器を指し、ビデオカードと LSI チップを区別しないで呼ばれることが多い。

4.2.2 GPU の歴史

1970 年代に入り大型のコンピュータのサイズダウンが行われ、ミニコンという種類が登場、低価格帯の CPU による一般向けのコンピュータ、パーソナルコンピュータ (パソコン) が登場した。世界最初のパソコンと呼ばれているアルテアにはモニタがなく、コンピュータの状態を LED で表示するにとどまった。モニタに対応したのアップルの Apple I、Apple II だった。当時のパソコンには、メモリ上にビデオ表示用の領域があり、後に Video RAM (VRAM) と呼ばれるようになった。VRAM の領域を書き換えることで画面に出力ができた。

1990 年代に入るとパソコンでも GUI の環境が整ってきた。GUI が発展すると以前の 640 × 400 ドットより広い描画範囲を必要とした。そのため、ユーザの間ではビデオカードを用いて描画領域を拡張させる動きがあった。Windows95 が発売のことには、ビデオカードが含まれるパソコンが標準となった。

1999 年に NVIDIA から GeForce256 が発売された。GeForce256 の特徴は 3D 演算用のハードウェアが搭載されたことである。それまで 3D 専用ワークステーションでしか行えなかった分野がパソコンでも行えるようになった。CPU が通常の演算を行うことから、グラ



図2 グラフィックボード

フィックの演算処理を行うハードウェアを GPU と呼ぶと NVIDIA が提案，その呼び名が定着した。

2001 年には GPU の固定機能パイプラインであるシェーダ処理がプログラム可能になった。しかし，アセンブラが基本として扱われたため敷居は非常に高かった。その後，シェーダのプログラムを書くための高級言語が登場，3D グラフィック以外の演算も行われるようになった。演算ユニットを汎用化させる統語柄シェーダーアーキテクチャが登場，GPU における汎用演算 GPGPU の発展，普及へとつながっていった。

2010 年に入ると CPU 分野ではマルチコアを搭載するだけでなく，画像出力専用回路として GPU コアを統合した製品の提供を行った。それまで行われていた画像データを外部メモリ空間へ転送するといった処理は不要になった。

4.2.3 GPU の処理方法

5 プログラム

本章では本論文の核となるプログラミングについて解説を行う.

5.1 JavaScript

5.2 WebWorker

5.3 OffscreenCanvas

6 実装したシミュレータ

6.1 開発したシミュレータ

6.2 実験方法

7 実験結果

8 結言

9 謝辞

本研究の遂行及び本論文の制作に当たり、助言をくださった須田研究室の仲間に深く感謝の意を表します。そして、本論文の制作に当たり多大なるご指導及びご助言をいただきました須田宇宙准教授に深く感謝いたします。

10 参考文献

11 付録 制作したプログラム