ĐẠI HỌC QUỐC GIA TPHCM

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

BỘ MÔN CÔNG NGHỆ TRI THỨC

Phương pháp toán trong phân tích dữ liệu thị giác

Đề tài: Tên Báo Cáo Gì Đấy

Môn học: Môn Học Gì Đấy

Sinh viên thực hiện:

Nguyễn Quốc Anh - 21127006

Giáo viên hướng dẫn:

GS. TS. Lý Quốc Ngọc

Ngày 16 tháng 10 năm 2024

Mục lục

Danh sách bảng

Danh sách hình vẽ

1 Lecture 1. Introduction to Mathematical Method in VDA

Introduction

the fourth industrial revolution computer vision

1.1 Intelligent Vision System

1.1.1 The concept of Intelligence

Some indications of intelligence that are of interest include:

- memory, recall, creativity
- computation, inference, recognition, prediction
- retrieval, localization & moving, reasoning

1.1.2 The intelligence levels of Vision System's output

1.1.3 The three basic levels of Vision Systems:

- Basic methods for data processing

  Q: Những khám phá nào ở cấp độ 1 mà làm thay đổi ngoạn mục cấp độ 2 và 3?

  A: Fast Fourier Transform (FFT) and Convolutional Neural Network (CNN).
- Single task processing
- Complex applications processing.

1.1.4 Some Intelligent Vision Systems

- Intelligent Transportation System (ITS)
- Intelligent Monitor System (IMS)
- Autonomous Vehicle System (AVS)
- Fault Inspection System (FIS)
- Disease Diagnosis System based on Imaging

- Harvesting System in Agriculture
- Intelligent Image-Video Retrieval

1.1.5 Learning Method

Everything advances slowly Ngta sẽ nghiên cứu các mô hình học máy để cải tiến hơn nữa.

- Supervised learning (semi-, self-)
- Unsupervised learning
- Reinforcement learning: học tăng cường, nổi lên thông qua AlphaGo, bây giờ ứng dụng trong xe tự hành, ChatGPT,... RL is a machine learning technique that focouses on training and algorithm following the cut-and-try approach. The algotirithm
  - The agent or the learner
  - The environment

  Examples:
  - any real-world problem where an agent must interact with an uncertain environment to meet a specific goal: robotics, AlphaGo, autonomous driving, logistics,...

  Benefits:
  - Artificial General Intelligence (AGI)
  - does not need a separate data collection step
- Continual learning: học
- Federated learning
- Deep learning
- Transfer learning

- Meta learning Deep neural networks can achieve great succes when presented wiht large datasets and sufficient computational resources. However, their ability to learn new concepts quickly is limited. It is one of the defining aspects of human intelligence (Jankowski, 2018). Meta learning is one approach to this issue by enabling the network to learn how to learn.

  - Image Classification
  - Facial Recognition and Face Antispoofing
  - Person-specific talking head generateion for unseen

Traditional Programming data -> computer -> results set of rules (proram)
Machine Learning data -> computer -> set of rules (model) results (Optional)
Machine learning as a field is "concerned with the question of how to construct computer programs that automatically improve with experience." - Tom Mitchell 1997 he presents the formal definition of machine learning as follows:
"A computer program is said to learn from **experience E** with respect to some **class of tasks T** and **performance measure P**, if its performance at tasks in T, as measured by P, improves with experience E."
There are three main machine learning paradigms:

- Supervised learning: learning properties of data using labelled data
- Unsupervised learning: learning properties of data using unlabelled data
- Reinforcement learning: learning properties of an environment through trial and error

learning Method
advanced deep neural network system advanced deep neural network architecture

- Le-Net, AlexNet, VGG, GoogleNet
- ResNet, SeNet, EfficientNet
- Graph Neural Network (GNN)
- Generative Adversarial Network (GAN)

- Vision Transformer (ViT)

Q: Có mạng học máy nào mới gần đây không? A:

1.2 Generative AI

1.2.1 Generative AI in Computer Vision

- Generative model
  - GAN
  - DIFFUSION
- Image-tasks
  - Text2Image, Image2Text
  - Style2Image
  - HumanBrainSignal2Image
- Video-tasks
  - Text2Video, Video2Text
  - Text2Animation
- Computer Graphics-tanks
  - Text23DScene
  - Text23DObjectAnimation

Q: Ví dụ về lĩnh vực, chủ đề mà Generative AI có thể hỗ trợ, không có không được A: Image2Text (e.g. tóm tắt video)

1.2.2 Diffusion Model

"Diffusion Models are a class of probabilistic generative models that turn noise to a representative data sample."

Using Diffusion models, we can generate images either conditionally or unconditionally.

1. Unconditional image generation simply means that the model converts noise into any "random representative data sample." The generation process is not controlled or guided, and the model can generate an image of any nature.

Q: Nếu như đánh context "flamingos standing on water, red sunset, pink-red water reflection" thì máy có generate được hình ảnh khác không?

A:

Q: mỗi lần tôi gõ cùng một câu thì nó ra một kết quả khác nhau hay giống nhau?

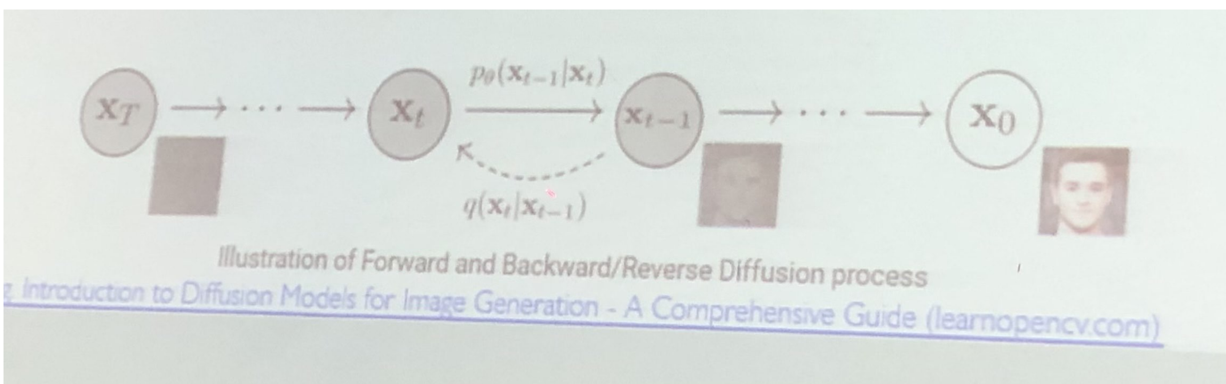
A: tuỳ vào model

Q: Nếu Intelligence System chỉ dừng ở mức thông minh mà không phải ở mức thông tuệ thì sẽ dừng ở lĩnh vực nào?

A:

An idea used in non-equilibrium statistical physics is that we can **gradually convert one distribution into another**. In 2015, Sohl-Dicktein et al., inspired by this, created "Diffusion Probabilistic models" or "Diffusion models" in short, building on this essential idea.

They build - **"A generative Markov chain which converts a simple known distribution (e.g., a Gaussian) into a target (data) distribution using a diffusion process."**



Hình 1: Illustration of Forward and Backward/Reverse Diffusion process

In 2015, Sohl-Dickstein et al. published paper "Deep Unsupervised Learning using Nonequilibrium Thermodynamics" and Diffusion models in deep learning were first introduced.

In 2019, Song et al. published a paper, "Generative Modeling by Estimating Gradients of the Data Distribution," using the same principle but a different approach.

In 2020, Ho et al. published the paper, now-popular "Denoising Diffusion Probabilistic Models" (DDPM for short).

After 2020, research in diffusion models took off. Much progress has been made in creating, training, and improving diffusion-based generative modeling in a relatively short time.
Some of the Diffusion-based Image Generation models that became famous over the past few months.
Some typical famous Diffusion-based Image Generation models include:

- **DALL-E 2** by OpenAI
- **Imagen** by Google
- **Stable Diffusion** by StabilityAI
- **Midjourney**

D and G play the following two-player minimax game with value function $V(G, D)$:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$

Discriminator output for real data x
Discriminator output for generated fake data G(z)
Min Max V(D,G) = Ex-Plate [logD(x)] + Ez-P2(2) [log(1 – D(G(z)))] G
Error from the discriminator model training
Error from the combined model training
Explainable Al inComputer Vision
Role of XAI in Computer Vision
The absence of explicability and transparency in certain areas is not invariably a problem since state- of-the-art models have an extremely high accuracy.
However, in areas such as autonomous cars, financial transactions and mainly medical applications, failures are unacceptable, considering that erroneous decisions can have disastrous con-sequences, such as the loss of human lives. Due to this fact, these application areas have extreme interest in explaining and interpreting each decision made by deep learning models.
It is possible to use Explaining Artifcial Intelligence to improve deep learning models performance.

1.3 Vision-Language Pre-trained Model

1.3.1 Vision-Language Pre-trained

Why is the Vision-Language Pre-trained Model necessary?

Most visual recognition studies rely heavily on crowd-labelled data in deep neural networks (DNNs) training, and they usually train a DNN for each single visual recognition task, leading to a laborious and time-consuming visual recognition paradigm.

To address the two challenges, Vision-Language Models (VLMs) have been intensively investigated recently, which learns rich vision-language correlation from web-scale image-text pairs that are almost infinitely available on the Internet and enables zero-shot predictions on various visual recognition tasks with a single VLM.

1.3.2 Text-Image Tasks

Model Architecture

Given an image-text pair, a VL model first extracts text features w = W1, , WN and visual features v = (V1, VM) via a text encoder and a vision encoder, respectively. Here, N is the number of tokens in a sentence, and M is the number of visual features for an image, which can be the number of image regions/grids/patches, depending on the specific vision encoder being used. The text and visual features are then fed into a multimodal fusion module to produce cross-modal representations, which are then optionally fed into a decoder before generating the final outputs.

Core VisionTasks

Q: A:

Dataset for VLM

2 Lecture 2. Metric Space

2.1 The Role of Metric Space in VDA

2.2 The Basic Concepts in Metric Space

2.2.1 Metric Space

Metric space (X, d) is space (X) together with a real-valued function d, $d : X \times X \to \mathbb{R}$, which measures the distance between pairs of points x and y in X.
d obeys the following axioms:

(i) $0 < d(x, y) < \infty \quad \forall x, y \in X, x \neq y$

(ii) $d(x, x) = 0 \quad \forall x \in X$

(iii) $d(x, y) = d(y, x) \quad \forall x, y \in X$

(iv) $d(x, y) \leq d(x, z) + d(z, y) \quad \forall x, y, z \in X$

Such a function d is called a metric.

2.2.2 Cauchy sequences

A sequence of points $\{x_n\}_{n=1}^{\infty}$ in a metric space X, d is called a Cauchy sequence if:

$$\forall \epsilon > 0, \exists N > 0 \text{ such that } d(x_n, x_m) < \epsilon \quad \forall n, m > N$$

2.2.3 Convergent sequence

A sequence of points $\{x_n\}_{n=1}^{\infty}$ is said to converge to a point $x \in X$ in metric space X, d if:

$$\forall \epsilon > 0, \exists N > 0 \text{ so that } d(x_n, x) < \epsilon \quad \forall n > N$$

$x \in X$, to which the sequence converges, is called the limit of the sequence, and we use the notation:

$$x = \lim_{n \to \infty} x_n$$

2.2.4 Complete Metric Space

Theorem 1. (*Convergent sequence & Cauchy sequence*)

A sequence of points $\{x_n\}_{n=1}^{\infty}$ in metric space X, d converges to a point $x \in X$, then $\{x_n\}_{n=1}^{\infty}$ is a Cauchy sequence.

Definition 4. (*Complete metric space*)

A metric space X, d is complete if every Cauchy sequence $\{x_n\}_{n=1}^{\infty}$ in X has a limit $x \in X$.

2.2.5 Metric space Hausdorff

Let X, d be a complete metric space. Then $\mathcal{H}(X)$ denotes the space whose points are the compact subsets of X, other than the empty set.

Definition 6. (*Metric in space Hausdorff*)

Let X, d be a complete metric space. Let $A, B \in \mathcal{H}(X)$. The Hausdorff distance between points A and B in $\mathcal{H}(X)$ is defined by:

$$h(A, B) = \max\{d(A, B), d(B, A)\}$$

where

$$d(A, B) = \max\{d(x, B) : x \in A\}$$

$$d(x, B) = \min\{d(x, y) : y \in B\}$$

Theorem 2. (*The completeness of metric space Hausdorff*)

Let X, d be a complete metric space. Then $(\mathcal{H}(X), h_d)$ is a complete metric space. Moreover, if $\{A_n \in \mathcal{H}(X)\}_{n=1}^{\infty}$ is a Cauchy sequence then:

$$A = \lim_{n\to\infty} A_n \in \mathcal{H}(X)$$

$$A = \{x \in X : \exists \text{ a Cauchy sequence } \{x_n \in A_n\} \to x\}$$

2.2.6 Contraction mapping and fixed point

Definition 1. (*Contraction mapping*)

A transformation $f : X \to X$ on a metric space X, d is called contractive or a contraction mapping if:

$$\exists s, 0 \leq s < 1 \text{ such that } d(f(x), f(y)) \leq s \cdot d(x, y) \quad \forall x, y \in X$$

Any such number s is called a contractivity factor for f.

Theorem 3. (*Contraction mapping*)

Let $f : X \to X$ be a contraction mapping on a complete metric space X, d. Then f possesses exactly one fixed point $x_f \in X$ and moreover for any point $x \in X$, the sequence $\{f^n(x)\} \to x_f$. That is:

$$\lim_{n \to \infty} f^n(x) = x_f \quad \forall x \in X$$

Theorem 4. (*Fixed point approximation*)

Let $f : X \to X$ be a contraction mapping on a complete metric space X, d with contractivity factor s. The fixed point x_f is approximated by the following expression:

$$d(f^n(x), x_f) \leq \frac{s^n}{1 - s} d(x, f(x)) \quad \forall x \in X$$

Theorem 5. (*Approximate x by fixed point*)

Let $f : X \to X$ be a contraction mapping on a complete metric space X, d with contractivity factor s, fixed point $x_f \in X$. Then:

$$d(x, x_f) \leq \frac{1}{1 - s} d(x, f(x)) \quad \forall x \in X$$

2.2.7 Contraction mapping on metric space Hausdorff

Lemma 1. (*Contraction mapping*)

Let $f : X \to X$ be a contraction mapping on a complete metric space X, d with contractivity factor s. Then $w : \mathcal{H}(X) \to \mathcal{H}(X)$ defined by:

$$w(B) = \{w(x) : x \in B\} \quad \forall B \in \mathcal{H}(X)$$

is a contraction mapping on $\mathcal{H}(X), h(d)$ with contractivity factor s.

Lemma 2. (*Contraction mapping sequence*)

Let X, d be a metric space. Let $\{w_n\}_{n=1}^{\infty}$ be contraction mappings on $\mathcal{H}(X), h(d)$ with contractivity factor for w_n denoted by s_n for each n. Define $W : \mathcal{H}(X) \to \mathcal{H}(X)$ by:

$$W(B) = w_1(B) \cup w_2(B) \cup \ldots \cup w_N(B)$$

$$= \bigcup_{n=1}^{N} w_n(B) \quad \text{for each } B \in \mathcal{H}(X)$$

Then W is a contraction mapping with contractivity factor:

$$s = \max\{s_n : n = 1, 2, \ldots, N\}$$

Theorem 6. (*Fixed set in metric space Hausdorff*)

Let $X; w_n, n = 1, 2, \ldots, N$ be an iterated function system with contractivity factor s. Then the transformation $W : \mathcal{H}(X) \to \mathcal{H}(X)$ defined by:

$$W(B) = \bigcup_{n=1}^{N} w_n(B) \quad \forall B \in \mathcal{H}(X)$$

is a contraction mapping on the complete metric space $\mathcal{H}(X), h(d)$ with contractivity factor s. Its unique fixed set, $A \in \mathcal{H}(X)$, obeys:

$$A = W(A) = \bigcup_{n=1}^{N} w_n(A)$$

$$A = \lim_{n \to \infty} W^n(B), \quad B \in \mathcal{H}(X)$$

Q: Ý nghĩa của tính chất này?

A: self-similarity (đặc tính tự tương tự)

$$\begin{aligned} A = W(A) &= \bigcup_{n=1}^{N} w_n(A) \\ &= \bigcup_{n=1}^{N} w_n(W(A)) \\ &= \bigcup_{n=1}^{N} \bigcup_{n=1}^{N} (w_n w_n(A)) \end{aligned}$$

→ tính chất tự phân hình (tính chất fractal)

Theorem 7. (*Approximate fixed set in metric space Hausdorff*)

Let $\{w_n\}_{n=1}^{N}$ be an iterated function system with contractivity factor s. Then the transformation $W : \mathcal{H}(X) \to \mathcal{H}(X)$ defined by:

$$W(B) = \bigcup_{n=1}^{N} w_n(B) \quad \text{for all } B \in \mathcal{H}(X)$$

$A \in \mathcal{H}(X)$ is a fixed set approximated (A không bao giờ có giá trị chính xác mà chỉ có giá trị xấp xỉ) by:

$$h(W^n(B), A) \leq \frac{s^n}{1-s} h(B, W(B)) \quad \forall B \in \mathcal{H}(X)$$

Theorem 8. (*Approximate O by fixed set*)

Let O be a subset of $\mathcal{H}(X)$. Let $\{w_n\}_{n=1}^{N}$ be an iterated function system with contractivity factor s. Then the transformation $W : \mathcal{H}(X) \to \mathcal{H}(X)$ defined by:

$$W(B) = \bigcup_{n=1}^{N} w_n(B) \quad \text{for all } B \in \mathcal{H}(X)$$

$A \in \mathcal{H}(X)$ is a fixed set of W,

$$h(O, A) \leq \frac{1}{1-s} h(O, W(O))$$

2.3 Applications of Metric Space in VDA

Q: Nếu d không thoả điều kiện trong phương trình iii: tính bắc cầu thì gây khó khăn gì? A:

Cho ví dụ về không gian metric

A: Cho không gian toạ độ ảnh, ta có: d(x,y): khoảng cách giữa 2 điểm x và y trong không gian Euclid, x = (x_1, x_2) và $y = (y_1, y_2)$ với $x_1, x_2, y_1, y_2 \in \mathbb{N}$

$$\begin{aligned}
0 < d(x,y) &= \sqrt{(x_1-y_1)^2+(x_2-y_2)^2} < \infty, \forall x \neq y, \forall x_1, x_2, y_1, y_2 \in \mathbb{N} \\
d(x,y) &= \sqrt{(x_1-y_1)^2+(x_2-y_2)^2} = 0 \Leftrightarrow x = y \\
d(x,y) &= \sqrt{(y_1-x_1)^2+(y_2-x_2)^2} = \sqrt{(x_1-y_1)^2+(x_2-y_2)^2} = d(y,x) \\
Let z &= (z_1, z_2) \in \mathbb{N}, \\
d(x,y) &= \sqrt{(x_1-y_1)^2+(x_2-y_2)^2} \\
&\leq d(x,z) + d(z,y) = \sqrt{(x_1-z_1)^2+(x_2-z_2)^2} + \sqrt{(z_1-y_1)^2+(z_2-y_2)^2}
\end{aligned}$$

$$\begin{aligned}
\Leftrightarrow (y_1-x_1)^2+(y_2-x_2)^2 &\leq (y_1-z_1)^2+(y_2-z_2)^2+(z_1-x_1)^2+(z_2-x_2)^2+ \\
&2\sqrt{(x_1-z_1)^2+(x_2-z_2)^2}\sqrt{(z_1-y_1)^2+(z_2-y_2)^2} \\
\Leftrightarrow (y_1-z_1+z_1-x_1)^2 &+ (y_2-z_2+z_2-x_2)^2 \\
&\leq (y_1-z_1)^2+(z_1-x_1)^2-2(y_1-z_1)(z_1-x_1) \\
&+ (y_2-z_2)^2+(z_2-x_2)^2+2(y_2-z_2)(z_2-x_2) \\
&\leq (y_1-z_1)^2+(z_1-x_1)^2+(y_2-z_2)^2+(z_2-x_2)^2 \\
\Leftrightarrow [(y_1-z_1)(z_1-x_1) &+ (y_2-z_2)(z_2-x_2)] \\
&\leq [(y_1-z_1)^2+(y_2-z_2)^2][(z_1-x_1)^2+(z_2-x_2)^2] \\
&\text{Bất đẳng thức Bunjakowski-Schwarz}
\end{aligned}$$

Q: Cho ví dụ về dãy Cauchy mà không phải dãy hội tụ A:

Q: Trong hằng hà sa số các ánh xạ, tại sao ánh xạ co được chú ý nhất? A:

Q: giải thích định lí ánh xạ co và ứng dụng của nó A:

Generative AI in Computer Vision

Generative Model

D and G play the following two-player minimax game with value function V (G, D):

Min Max V(D,G) = Ex-pan [logD(x)] + E-P(z) [log(1-D(G(z)))]

Discriminator output for real data x

Discriminator output for generated fake data G(z)

Min Max V(D,G) = Ex-Plate [logD(x)] + Ez-P2(2) [log(1 – D(G(z)))] G

Error from the discriminator model training

Error from the combined model training

HOHUI Quốc Ngpo 12023

57

2.4 Lecture 3. Apply metric space to VDA

2.4.1 Drawing based on Fractal geometry

3.1.1. Computing Fractals from Iterated Function Systems

Consider IFS { $\mathbb{R}^2$, w_n: $n = 1, 2, \ldots, N$ }

$$w_i \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix} A_i x + t_i$$

p_i associated with w_i

$p_i \approx \frac{|\det A_i|}{\sum_{i=1}^{N} |\det A_i|}$

$N = \sum_{i=1}^{N} \det A_i = \sum_{i=1}^{N} (a_i d_i - b_i c_i)$

3.1. Drawing based on Fractal geometry

Deterministic Algorithm

Let, be an IFS,

Compute directly a sequence of sets,

starting from an initial set $A \subseteq \mathbb{R}^2$

$A_{n+1} = \bigcup_{j=1}^{N} w_j(A_n)$, $n = 1, 2, \ldots$

Sequence $\{A_n\}$ converges to the attractor A of the IFS in the Hausdorff metric.

3.1. Drawing based on Fractal geometry

Deterministic Algorithm

1. Initialize $s[M, M]$, $t[M, M]$

2. Setup the values of IFS

$a[1] = 0.5; b[1] = 0; c[1] = 0; d[1] = 0.5; e[1] = 1; f[1] = 1$

$a[2] = 0.5; b[2] = 0; c[2] = 0; d[2] = 0.5; e[2] = 50; f[2] = 1$

$a[3] = 0.5; b[3] = 0; c[3] = 0; d[3] = 0.5; e[3] = 50; f[3] = 50$

3. Input the initial set $A(0)$ into $t[M, M]$

4. Repeat

5. For $i = 1$ to M /Apply W to $A(n)$ to make $A(n+1)$ in $s[i, j]$/

6. For $j = 1$ to M

7. If $t[i, j] = 1$ then

8. $s[a[1] * i + b[1] * j + e[1], c[1] * i + d[1] * j + f[1]] = 1$

$s[a[2] * i + b[2] * j + e[2], c[2] * i + d[2] * j + f[2]] = 1$

9. $s[a[3] * i + b[3] * j + e[3], c[3] * i + d[3] * j + f[3]] = 1$

10. End if

11. End /for j/

12. End /for i/

PGS.TS. Lý Quốc Ngọc

3.1. Drawing based on Fractal geometry

Deterministic Algorithm

13. For $i = 1$ to M

14. For $j = 1$ to M

15. $t[i, j] = s[i, j]$ /Put $A(n+1)$ into the array $t[i, j]$/

16. $s[i, j] = 0$ /Reset the array $s[i, j]$ to 0/

17. If $t[i, j] = 1$ then

18. setpixel(i, j) /Plot $A(n+1)$/

19. End if

20. End /for j/

21. End /for i/

22. Until $A(n+1) = W(A(n+1))$

PGS.TS. Lý Quốc Ngọc

3.1. Drawing based on Fractal geometry

Random Iteration Algorithm

Let be an IFS, where has been $1, 2, \ldots, N$ assigned to i, $\sum p_i = 1$

Choose $x \in X$ and then choose recursively, independently,

$x_n \in \{w_1(x_{n-1}), w_2(x_{n-1}), \ldots, w_N(x_{n-1})\}$, $n = 1, 2, 3, \ldots$

Where the probability of the event $x_n = w_i(x_{n-1})$ is p_i

Sequence $\{x_n\}$ be constructed converges to the attractor of

the IFS.

PGS.TS. Lý Quốc Ngọc

3.1. Drawing based on Fractal geometry

Random Iteration Algorithm

1. Initialize $s[M, M]$, $t[M, M]$

2. Setup the values of IFS

$a[1] = 0.5; b[1] = 0; c[1] = 0; d[1] = 0.5; e[1] = 1; f[1] = 1$

$a[2] = 0.5; b[2] = 0; c[2] = 0; d[2] = 0.5; e[2] = 50; f[2] = 1$

$a[3] = 0.5; b[3] = 0; c[3] = 0; d[3] = 0.5; e[3] = 50; f[3] = 50$

3. $x = 0; y = 0; numits = N$ /Initialize (x, y) and the number of iterations/

4. For $n = 1$ to $numits$ /Random Iteration/

5. $k = \text{int}(3 * \text{rnd} - 0.0001) + 1$ /Choose one of the numbers 1,2,3/

6. $newx = a[k] * x + b[k] * y + e[k];$

7. $newy = c[k] * x + d[k] * y + f[k];$

8. $x = newx;$

9. $y = newy;$

10. setpixel(x, y);

11. End /for n/

2.5 Fractal image compression

Method

Based on theorem of contraction mapping sequence in

metric space Hausdorff

Let the Original image be O

The compression image will be $\{w_n\}$

The ideal uncompressed will be A

The real uncompressed be $W(B)$

PGS.TS. Lý Quốc Ngọc

3 Chapter 3. Vector Space

3.1 The role of vector space in VDA

3.2 The basic concepts in vector space

3.3 Applications of vector space in VDA

4 Chapter 4. Optimization Method

4.1 The role of optimization method in VDA

4.2 Unconstrained optimization method

4.3 Constrained optimization method

4.4 Applications of optimization method in VDA

5 Chapter 5. Method of Solving a System of Equations

5.1 The role of system of equations in VDA

5.2 Method of solving system of linear equations

5.3 Method of solving system of non-linear equations

5.4 Applications of system of equations in VDA

6 Chapter 6. Method of Solving Partial Differential Equations

6.1 The role of PDE in VDA

6.2 Method of solving PDE

6.3 Applications of PDE in VDA

7 Chapter 7. Deep Learning