

Herond Robaina Salles

*Clustering de imagens via redes neurais de  
Kohonen associadas a momentos  
invariantes*

Campos dos Goytacazes/RJ

2012

Herond Robaina Salles

*Clustering de imagens via redes neurais de  
Kohonen associadas a momentos  
invariantes*

Monografia apresentada ao Curso de Graduação em Ciência da Computação da Universidade Estadual do Norte Fluminense Darcy Ribeiro como requisito para obtenção do título de Bacharel em Ciência da Computação, sob orientação da Prof<sup>a</sup>. Annabell Del Real Tamariz, DrSc.

Tutor: Annabell Del Real Tamariz, DrSc.

UNIVERSIDADE ESTADUAL DO NORTE FLUMINENSE DARCY RIBEIRO

Campos dos Goytacazes/RJ

2012

”E posto que se infligiram inutilmente ao corpo social tantos sistemas, que se termine por onde se deveria ter começado: que se rejeitem os sistemas; que se coloque, por fim, a Liberdade à prova - a Liberdade, que é um ato de fé em Deus e em sua obra.”

*Frederic Bastiat*

## Agradecimentos

AGRADECIMENTOS AQUI.

## *Lista de Figuras*

1	Diferença entre uma imagem vetorial e uma imagem <i>bitmap</i> . Uma imagem <i>bitmap</i> perde a qualidade quando ampliada, o que não ocorre com uma imagem vetorial . . . . .	14
2	Duas imagens idênticas porém espelhadas. . . . .	15
3	Exemplos de transformações de rotação, translação e escala sobre uma imagem, e como elas não alteram a essência das formas presentes na imagem original. . . . .	16
4	Imagem com o primeiro plano e o plano de fundo destacados. . . . .	19
5	Imagens visualmente semelhantes mas com relativa diferença nos valores dos momentos devido as grande diferença de tons no primeiro plano. . . .	19
6	Imagem binarizada com limiar definido pelo método de Otsu. . . . .	21
7	Esquema do processo de extração de características. . . . .	22
8	Organização das classes em um mapa auto organizável, coesão interna entre os elementos e isolamento externo entre as classes. . . . .	24
9	Modelo de neurônio. . . . .	25
10	Representação de uma rede de Kohonen. . . . .	26
11	Esquema detalhado de uma rede de Kohonen. . . . .	27
12	Representação da malha de distâncias topográficas entre os neurônios . . .	32
13	Distâncias $d_x$ , $d_y$ e $d_{xy}$ entre o neurônio $b_{x,y}$ e seus vizinhos. . . . .	33
14	Organização da U-Matriz em relação a grade de neurônios. . . . .	34
15	Legenda. . . . .	34
16	As três possíveis situações na passagem do nível $h_{min}$ para $h_{min+1}$ . . .	36
17	Legenda. . . . .	39

18	Variação do erro durante as 533 iterações necessárias para completar o treinamento da rede. . . . .	46
19	Variação da U-matriz ao longo do treinamento da rede. . . . .	46
20	Variação do erro durante as 533 iterações necessárias para completar o treinamento da rede. . . . .	47

## *Lista de Códigos*

## *Resumo*

RESUMO AQUI.



# *Sumário*

<b>Lista de Figuras</b>	<b>2</b>
<b>Resumo</b>	<b>5</b>
<b>1 Introdução</b>	<b>8</b>
1.1 Tema . . . . .	9
1.2 Objetivo geral . . . . .	10
1.3 Objetivos específicos . . . . .	10
1.4 Justificativa . . . . .	11
1.5 Organização do trabalho . . . . .	11
<b>2 Descritores de imagens</b>	<b>13</b>
2.1 Conceitos introdutórios sobre imagens digitais . . . . .	13
2.2 Simplificação de imagens e extração de características . . . . .	14
2.3 Momentos invariantes como descritores de imagens . . . . .	15
2.3.1 Formulação matemáticas dos momentos invariantes . . . . .	16
2.4 Binarização de imagens . . . . .	18
2.5 Método de Otsu . . . . .	20
2.6 Resumo do processo de extração de características . . . . .	21
<b>3 Rotulação das imagens</b>	<b>23</b>
3.1 Mapas auto organizáveis . . . . .	23
3.2 Redes de Kohonen como mapa auto organizável . . . . .	24

3.3	Características gerais de uma rede neural de Kohonen . . . . .	26
3.3.1	Topologia de uma rede de Kohonen . . . . .	26
3.3.2	Treinamento da rede . . . . .	27
3.4	Normalização dos dados de entrada da rede . . . . .	29
3.5	A rede neural de Kohonen e o agrupamento das imagens . . . . .	31
3.5.1	Matriz de distâncias unificadas . . . . .	31
3.5.2	Transformada <i>watershed</i> para rotulação automática da U-Matriz . .	34
3.6	Resumo do processo de classificação das imagens e do <i>clustering</i> como um todo . . . . .	37
<b>4</b>	<b>Testes empíricos e análise dos resultados</b>	<b>40</b>
4.1	Roteiro e fundamentação do teste proposto . . . . .	40
4.2	Especificação do conjunto de imagens e das classes de controle . . . . .	40
4.3	Considerações sobre a implementação e a plataforma de execução . . . . .	45
4.4	Treinamento da rede, convergência do erro e variação da U-matriz . . . . .	45
4.5	Tempo de execução . . . . .	47
4.6	Disposição das imagens e rotulação das imagens no mapa . . . . .	47
4.7	Avaliação geral dos resultados . . . . .	48
<b>5</b>	<b>Conclusões</b>	<b>49</b>
	<b>Referências Bibliográficas</b>	<b>50</b>

# 1 *Introdução*

A partir do momento que um grande volume de dados passou a ser armazenado em meios digitais, técnicas automáticas para classificação ou agrupamento (*clustering*) foram desenvolvidas com o propósito de descobrir que tipo de informação estes dados “escondem”, isto é, qual o tipo de relação, muitas vezes imperceptível aos usuários, os dados possuem entre si. Estas relações são, em muitos casos, imprescindíveis na tomada de decisões, havendo assim um forte interesse na criação e aperfeiçoar destas técnicas, seja para melhorar o tempo de execução, aumentar sua acurácia ou ainda ampliar a quantidade de informação que pode ser inferida.

Inicialmente estas técnicas foram empregadas em dados puramente numéricos, destacando-se informações bancárias, de geoprocessamento e informações sobre contratantes de planos de seguro. Contudo, assim que imagens passaram a ser processadas e armazenadas computacionalmente, surgiram também técnicas destinadas a agrupá-las, com o mesmo objetivo, descobrir inter-relações entre as imagens através do grupo para qual eram classificadas. Porém, com a popularização da internet e das câmeras digitais diversos novos modos para aquisição e compartilhamento de imagens surgiram, e com isso, as técnicas de classificação adquiriram ou novo objetivo, passaram a ser utilizadas na recuperação de imagens baseada em seu conteúdo visual, pois sabendo a que grupo uma determinada imagem pertence, pode-se utilizar as imagens que estão no mesmo grupo como resposta de uma consulta, ou no mínimo, como conjunto para uma busca direcionada, evitando que a busca seja executada por todo conjunto de imagens.

Há, entretanto, uma dificuldade adicional ao se classificar imagens. De modo geral, dados numéricos possuem um vasto ferramental teórico para definir objetivamente o quanto são parecidos, isto é, o gral de semelhança entre eles. Não é possível, porém, definir uma medida objetiva para o quanto semelhante são duas imagens, isto porque os dados numéricos representam, por si só, uma magnitude objetiva de uma determinada medida, as imagens no entanto, são representações computacionais da visão humana, e como tal, sua representação, aqui tomado como significado, está intimamente relacionada

a forma como os seres humanos, subjetivamente, a identificam. Duas pessoas podem divergir significativamente quanto a semelhança entre duas imagens, pois cada uma pode se apegar a uma interpretação cognitiva diferente, e nenhuma das duas estaria errada quanto a essa medida.

A forma encontrada para contornar este problema foi extrair das imagens determinados valores numéricos, chamado de descritores, e estes, ao invés das imagens, são classificados. Existem diversos tipos de descritores, baseados em diversas premissas sobre como se deve analisar a semelhança entre as imagens, de modo que não são totalmente arbitrários e possibilitam que técnicas de classificação convencionalmente utilizadas para outros tipos de dados também sejam utilizadas com imagens.

Paralelo ao problema dos descritores, determinadas técnicas de classificação impõe rígidas regras ao modo como as classes são determinadas, em particular, precisam que o usuário defina a quantidade de classes que serão encontradas. Isto pode fazer sentido para uma grande quantidade de classificações, como por exemplo, para classificar imagens como “quentes” ou “frias”, para identificar determinados gestos da mão ou expressões faciais, contudo, para os casos onde não sabemos que tipo de relações desejamos encontrar, este tipo de abordagem é ineficiente pois transfere ao usuário a obrigação de definir algo que, dado a própria natureza do problema, ele não tem capacidade de saber, ou no máximo, tem uma capacidade limitada. Contudo, técnicas mais sofisticadas não exigem este tipo de informação, trazendo para seu próprio escopo a tarefa de inferir a quantidade de classes de acordo com a natureza dos dados.

Por fim, classificar imagens significa agrupá-las segundo determinadas características de similaridade; porém, o mapeamento deve ocorrer de tal forma que cada classe forneça, essencialmente, as mesmas informações sobre as imagens que a compõem. As técnicas de classificação podem fornecer um resumo conciso do conteúdo das imagens, podendo ser utilizado para diferentes tarefas relacionadas com a gestão de banco de dados de imagens.

## 1.1 Tema

No geral a classificação de imagens segue as abordagens convencionais para classificação de dados, a diferença consiste em que, as imagens, devido a sua representação complexa, não são avaliadas diretamente pelos algoritmos, mas sim um conjunto de descritores devidamente calculados. Estas técnicas de classificação geralmente necessitam do número de classes que serão identificados, as mais sofisticadas exigem apenas um número mínimo.

Este trabalho pretende modelar uma técnica de classificação de imagens centrada nas redes neurais de Kohonen, que por isto não necessite da indicação explícita do número de classes que serão encontradas, de modo que esta informação é um de seus resultados e não um parâmetro necessário para sua aplicação.

A técnica é descrita pela apresentação formal das teorias que a compõe, bem como suas devidas justificativas. É analisado as características, fundamentos matemáticos e concepções de alto nível.

A técnica aqui apresentada não dispensa a necessidade de descritores, deste modo os momentos invariantes de Hu são utilizados para este propósito. Uma análise de todo o processo de extração de características é feito, o que insere no contexto do trabalho conceitos de tratamento de imagens, em especial binarização e o método de Otsu.

A rede de Kohonen é uma técnica emergente de aprendizado de máquina direcionada a classificação de dados que se apoia no conceito de mapas auto organizáveis para distribuir os elementos conforme seu gral de similaridade. Associadas as redes de Kohonen estão a U-matriz e a transformada de *watershed*, que são utilizadas em conjunto para determinar as classes para quais as imagens serão rotuladas.

## 1.2 Objetivo geral

Definir uma técnica de classificação de imagens onde os grupos são definidos de acordo com formas (silhuetas) presentes nas imagens e que não haja a necessidade do usuário informar a quantidade de classes a serem identificadas, ao contrário, as classes devem ser um resultado emergente do processo.

## 1.3 Objetivos específicos

Associados ao objetivo geral, este trabalho também pretende:

- Investigar os métodos que compõe a técnica proposta, no caso, limiarização/binarização de imagens, método de Otsu para limiares globais, redes neurais de Kohonen, U-matriz e transformada de *watershed*;
- Implementar a técnica em ANSI C;

- Elaborar um conjunto de testes para validar os resultados da técnica proposta, identificando suas limitações caso existam.

## 1.4 Justificativa

Como já foi argumentado no início deste capítulo, as técnicas de agrupamento de dados são importantes em diversas áreas, e em particular, a classificação de imagens tem se destacado recentemente em virtude do forte crescimento da internet, e com ela, o surgimento de uma nova modalidade de obtenção e compartilhamento de imagens; e também, pela própria sofisticação de determinados serviços de análise de imagens. Levando em consideração estes fatos, o desenvolvimento e aperfeiçoamento de técnicas de classificação onde é necessário o mínimo de interação humana, mais especificamente, onde as classes não precisem ser definidas pelo usuário, suscita um grande interesse de pesquisa.

A ausência da necessidade de se definir a quantidade de classes não é apenas uma questão de comodidade, mas também, em muitos casos, uma questão de racionalidade metodológica, afinal, a definição do número de classes supõe certo conhecimento sobre o grupo de imagens que se deseja classificar, contudo, na maioria dos casos este conhecimento não existe, principalmente quando a quantidade de imagens é muito grande, logo, uma técnica sem esta limitação é imprescindível para estas situações.

## 1.5 Organização do trabalho

Esta monografia é dividida em 5 capítulos, onde inicialmente busca-se apresentar o problema e o panorama geral onde o tema está situado, bem como o que motivou sua investigação e resolução.

No Capítulo 2 uma breve introdução sobre a representação de imagens digitais é feita, contudo, o objetivo do capítulo é descrever a extração de características das imagens e os algoritmos envolvidos neste processo.

No Capítulo 3 é feita a formalização do processo de classificação propriamente dito, assim como a contextualização e justificativa de todos os formalismos adotados na sua definição. O Capítulo se concentra especialmente nas redes neurais de Kohonen e em como ela pode ser adotada na classificação das imagens e na geração automática das classes.

Tanto no Capítulo 2 quanto no 3, ao final, são apresentados um resumo e um dia-

grama para facilitar a compreensão do funcionamento global e da inter-relação entre os componentes de cada etapa da classificação.

O Capítulo 4 descreve o roteiro de testes utilizados para validar a técnica, assim como os resultados obtidos, também são apresentadas algumas análises para dar suporte e interpretação aos resultados.

Por fim, o Capítulo 5 discorre sobre as conclusões acerca de tudo que foi discutido, algumas avaliações e propostas de trabalhos futuros também são indicadas neste capítulo.

O código fonte com alguns comentários adicionais sobre sua implementação pode ser encontrado nos anexos do trabalho.

## 2 *Descritores de imagens*

Tendo em vista que as imagens, devido a grande quantidade de informações contidas em sua representação, não servem como entrada para uma rede de Kohonen, é necessário extrair delas um conjunto resumido e mensurável de características que possam servir de entrada para rede. É sobre a extração deste conjunto de características que trata este capítulo. Serão abordados os descritores escolhidos para caracterizar uma imagem e também todo o tratamento que a imagem sofre até que estes descritores possam ser extraídos.

### 2.1 Conceitos introdutórios sobre imagens digitais

Antes de iniciar qualquer discussão a respeito da caracterização e agrupamento de imagens é necessário discorrer sobre o modo com as imagens são representadas computacionalmente, os termos comumente empregados nestas representações e, em particular, estabelecer quais destas representações serão adotadas neste trabalho.

Existe basicamente duas formas de representação computacional de imagens, mapa de bits (*bitmap*): uma matriz de pontos (*pixels*) que representam cores; ou vetoriais: um conjunto de descrições de formas geométricas, cores e texturas que, precisamente por serem equações vetoriais ou transformações matemáticas, não perdem a qualidade quando redimensionadas ou rotacionadas; a comparação entre estas duas representações pode ser observada na Figura 1.



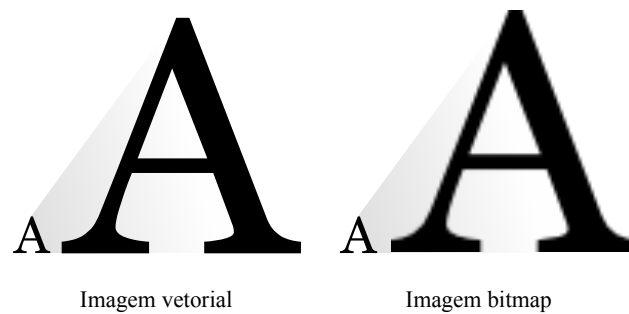


Figura 1: Diferença entre uma imagem vetorial e uma imagem *bitmap*. Uma imagem *bitmap* perde a qualidade quando ampliada, o que não ocorre com uma imagem vetorial

Quanto a cores, existem diversas padrões de representação, a codificação RGB (sigla para *Red*, *Green*, *Blue*) é a mais comum e define três bytes para armazenar, respectivamente, o vermelho, o verde e o azul, cada uma sendo um inteiro na faixa de 0 a 255. Outros padrões de representação são HLS (sigla para *Hue*, *Lightness*, *Saturation*), HSB (sigla para *Hue*, *Saturation*, *Brightness*), HSV (sigla para *Hue*, *Saturation*, *Value*), Hunter Lab, CIE 1976 Lab e CMYK (sigla para *Cian*, *Magenta*, *Yellow*, *Black*), este último utilizado em mídias impressas.

Embora uma imagem *bitmap* seja armazenada na RAM com todos os *pixels* é comum, por uma questão de economia de espaço e tempo de transmissão, a compressão destes arquivos. Entre todos os formatos de compressão os mais conhecidos são o GIF (*Graphics Inter-change Format*), o JPEG (*Joint Photographic Experts Group*) e o PNG (*Portable Network Graphics*).

Neste trabalho as imagens sempre serão *bitmap*, com as cores codificadas no padrão RGB e comprimidas no formato JPEG. As imagens serão tratadas como equações, notacionadas na forma  $f(x, y)$ , onde  $x$  e  $y$  são inteiros e indicam a posição de um *pixel* específico, e os pixels são interpretados como tuplas na forma  $(r, g, b)$ , onde  $r$ ,  $g$  e  $b$  pertencem ao subintervalo inteiro de 0 a 255 e representam as cores vermelha, verde e azul respectivamente.

## 2.2 Simplificação de imagens e extração de características

Qualquer método de agrupamento depende sensivelmente do critério de semelhança adotado nas comparações entre os elementos, será esse critério que, basicamente, determinará a classe de cada elemento. O critério de semelhança deve ser baseado em alguma carac-

terística mensurável e comparáveis entre si, ou seja, deve haver uma forma de se estabelecer a distância entre diferentes valores desta característica. Esta distância determinará a semelhança entre os elementos, onde quanto mais próximos mais semelhantes.

Em imagens existem diversas características que servem como critérios de semelhança, do ponto de vista da percepção humana, estas características são comumente ligados as cores, texturas ou formas presentes na imagen, ou ainda, a uma combinação delas. Em relação as cores, medidas de histograma são as mais populares; em texturas é comum a utilização de momentos do histograma de brilho, matriz de co-ocorrência, granulometria e informações do espectro de Fourier; para formas se destacam os algoritmos de detecção de formas de interesse e os momentos invariantes; uma abordagem mista de cores e formas é possível através de modelos de misturas gaussianas.

### 2.3 Momentos invariantes como descritores de imagens

Supondo que uma forma particular esteja presente numa imagem  $A$ , e que outra bastante parecida esteja presente numa imagem  $B$ , e que em relação a  $A$  a forma em  $B$  está invertida, ou, para o exemplo ficar mais claro, que esta forma seja a silhueta de um rosto, que em  $A$  está virado a esquerda e em  $B$  virado a direita, como indicado na Figura 2, é um objetivo particular deste trabalho que ambas as imagens possuam descritores (características extraídas) bastante semelhantes, senão idênticos; afinal, em termos perceptivos, ou seja, em termos de significado que um observador atribui as imagens, neste exemplo  $A$  e  $B$ , ambas possuem a figura de um rosto e estar cada um virado numa direção é uma característica marginal e não deve influenciar no agrupamento.



Figura 2: Duas imagens idênticas porém espelhadas.

O mesmo pode ser dito para rotação, traslação e escala de formas em diferentes ima-

gens, o que se deseja é a forma em si, algo como seu protótipo, independente destas transformações, como indica a Figura 3. A pretensão é, ao se descartar estas transformações, simular o que aparentemente é o comportamento natural de um indivíduo ao, sem ajuda do computador, categorizar e agrupar imagens.

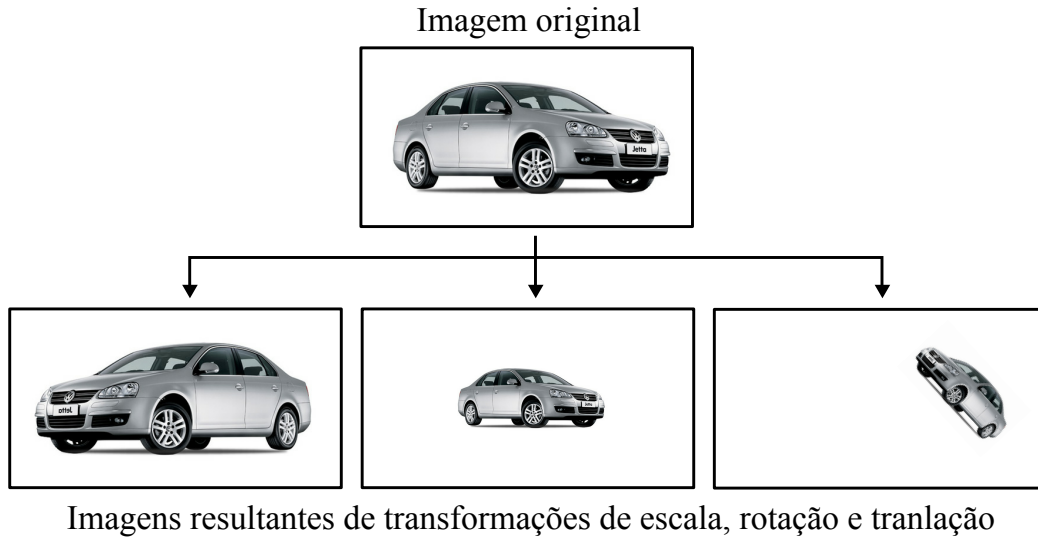


Figura 3: Exemplos de transformações de rotação, translação e escala sobre uma imagem, e como elas não alteram a essência das formas presentes na imagem original.

Um conjunto de descritores atende aos propósitos indicados acima, são os descritores de Hu, mais comumente chamados de momentos invariantes. Os momentos invariantes são um conjunto de sete descritores reais que independem de rotação, translação ou escala, isto é, quando aplicados a uma forma qualquer retornará os mesmos valores se aplicado a outra forma resultante de uma das três transformações citadas, ou até mesmo de uma combinação delas.

### 2.3.1 Formulação matemáticas dos momentos invariantes

Passemos então agora para formalização matemática desses momentos.

O momento bidimensional de ordem  $(p + q)$  é dado pela equação 2.1:

$$m_{pq} = \iint x^p y^q f(x, y) dx dy, p, q \in \quad (2.1)$$

A equação num domínio discreto, pode ser reescrita na forma:

$$m_{pq} = \sum_{x,y} x^p y^q f(x, y), p, q \in \quad (2.2)$$

A massa total da função  $f(x, y)$  é determinado pelo momento  $m_{00}$ , conforme a equação 2.3:

$$m_{pq} = \sum_{x,y} f(x, y), p, q \in \quad (2.3)$$

Existe um ponto no qual a aplicação pontual da massa total gera o mesmo momento que a massa distribuída, este ponto é dito centroide de  $f(x, y)$  e suas coordenadas  $x$  e  $y$  são dadas pela equação 2.4:

$$\bar{x} = \frac{1}{m_{00}} \sum x f(x, y) = \frac{m_{10}}{m_{00}} \quad (2.4a)$$

$$\bar{y} = \frac{1}{m_{00}} \sum y f(x, y) = \frac{m_{01}}{m_{00}} \quad (2.4b)$$

O momento central é obtido se deslocando a imagem para o centroide, da seguinte forma:

$$\mu_{pq} = \sum_{x,y} (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (2.5)$$

Ainda é necessário normalizar o momento para que os valores resultantes não sejam extremos a ponto de serem ignorados pelo sistema de reconhecimento de padrões. O momento central de ordem  $(p + q)$  normalizado é obtido dividindo o momento central de  $y$  mesma ordem por um fator definido por  $\mu_{00}^\gamma$ , conforme indicado pela equação 2.6:

$$\gamma = 1 + \frac{p + q}{2} \quad (2.6a)$$

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma} \quad (2.6b)$$

A partir dessas equações são estabelecidos sete momentos invariantes à translação, rotação e escala, chamados de momentos de Hu, ou descritores de Hu. São eles:

$$\varphi_1 = \eta_{20} + \eta_{02} \quad (2.7a)$$

$$\varphi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (2.7b)$$

$$\varphi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (2.7c)$$

$$\varphi_4 = (\eta_{30} + \eta_{12})^2 + (3\eta_{21} + \eta_{03})^2 \quad (2.7d)$$

$$\begin{aligned} \varphi_5 = & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (2.7e)$$

$$\begin{aligned} \varphi_6 = & (\eta_{20} - \eta_{02}) [(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ & + 4\eta_{11}(\eta_{30} - \eta_{12})(\eta_{21} + \eta_{03}) \end{aligned} \quad (2.7f)$$

$$\begin{aligned} \varphi_7 = & (3\eta_{21} - \eta_{30})(\eta_{30} + \eta_{12}) [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & + (3\eta_{12} - \eta_{03})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (2.7g)$$

Observe que os momentos são definidos para um ponto de valor discreto, isto implica que devemos abandonar qualquer descrição vetorial de cores, neste caso, devemos passar uma cor do formato RGB para seu tom de cinza. Neste trabalho o tom de cinza para uma cor RGB é o valor médio para os canais vermelho, verde e azul.

## 2.4 Binarização de imagens

Mesmo que os momentos invariantes sejam, a princípio, bons descritores, eles não podem ser extraídos sem que a imagem tenha passado por algumas transformações. Estas transformações não são obrigatórias, isto é, não são restrições necessárias a aplicação dos momentos, mas são transformações que fazem sentido no processo de agrupamento, mais especificamente, no subprocesso de extração de características relevantes.

É perfeitamente válido supor que nem todas as *pixels* de uma imagem são relevantes, ou no mínimo, que determinados *pixels* são mais relevantes que outros, estes *pixels* mais relevantes podem ser interpretados como regiões de interesse, isto é, regiões que despertam maior atenção dos observadores. Em suma, podemos dividir a imagem em duas regiões, uma de interesse chamada de primeiro plano (*foreground*) e outra que pode ser negligenciada chamada plano de fundo (*background*), como no exemplo da Figura 4. A separação entre essas duas regiões é chamado de limiarização, ou ainda, remoção de fundo.

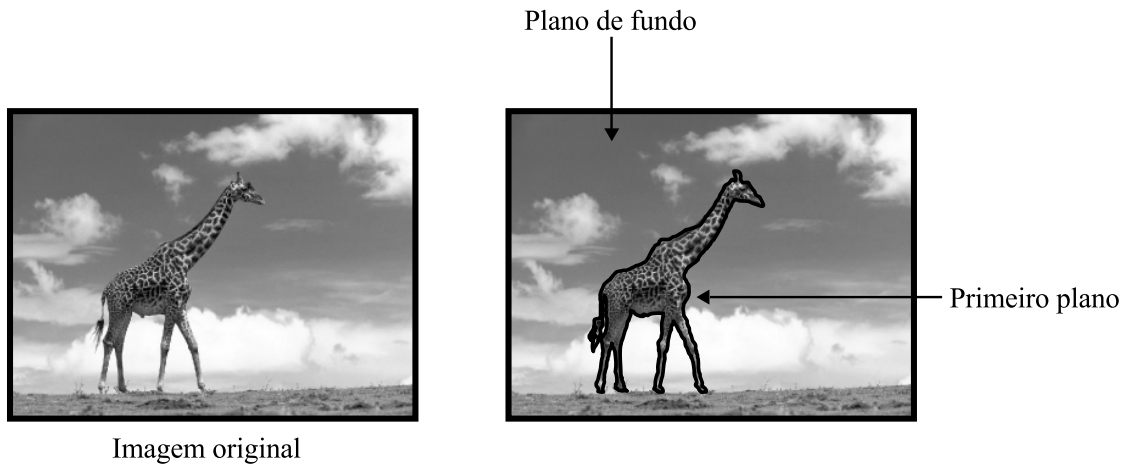


Figura 4: Imagem com o primeiro plano e o plano de fundo destacados.

Extraír da imagem os momentos invariantes apenas do primeiro plano torna os descritores mais interessantes para classificação, afinal, os valores ficam restritos apenas a região de maior interesse, sendo o plano de fundo totalmente ignorado na extração destas características.

Outro ponto a ser considerado, como visto na seção anterior, é que a extração dos momentos depende da intensidade de cada *pixel*, de modo que uma variação na intensidade de um *pixel* interfere no resultado dos momentos. Como agora apenas o primeiro plano é aplicado na extração, apenas as variações de intensidade nesta região são consideradas; contudo, estas variações podem em determinadas ocasiões gerar momentos muito distintos para regiões que, morfologicamente, são bem parecidas. Suponha o caso de, por exemplo, duas imagens que no primeiro plano apresentam a figura de uma flor, como na Figura 5, na primeira a flor tem a coloração clara, e na segunda escura, ao aplicar o momento sobre estas duas imagens, mesmo que tenham uma forma bem parecida, teremos resultados significativamente diferentes para os momentos das duas. É desejável eliminar este tipo de discrepância, isto é possível tornando todas as informações da primeiro plano homogêneas, ou seja, fazer com que cada *pixel* do primeiro plano tenha o mesmo peso para extração dos momentos. Esta homogenização sobre uma imagem já limiarizada é chamada de binarização, isto porque teremos duas regiões, uma irrelevante onde cada *pixel* terá o valor nulo, e outra relevante onde cada *pixel* terá seu máximo valor.

IMAGEM AQUI.

Figura 5: Imagens visualmente semelhantes mas com relativa diferença nos valores dos momentos devido as grande diferença de tons no primeiro plano.

Binarizar uma imagem, o que implicitamente também implica em limiarizá-la, é um processo bem simples e pode ser feito apenas como base no histograma. O que se deseja é anular todos os *pixels* abaixo de um limiar e potencializar os que estão acima dele. Como indicado no Algoritmo 1:

---

**Algorithm 1:** Binarização de uma imagem

---

**Entrada:**  $f(x, y)$  ,  $l$

**início**

**para cada**  $p \in f(x, y)$  **faça**

**se**  $p < l$  **então**

$p \leftarrow 0$

**senão**

$p \leftarrow 255$

**fim se**

**fim para cada**

**fim**

---

Contudo, o Algoritmo 1 não indica como definir o limiar ótimo, isto é, aquele que melhor separa o primeiro plano do plano de fundo, esta operação é realizada, neste trabalho, através do método de Otsu, descrito na próxima seção.

## 2.5 Método de Otsu

O método de Otsu é um método de *thresholding* global, isto é, o valor obtido é uma constante, para escolha do melhor limiar. A base deste método é sua interpretação do histograma como uma função de densidade de probabilidade discreta, da seguinte maneira:

$$p_r(r_q) = \frac{n_q}{n}, q = 0, 1, 2, \dots, L - 1 \quad (2.8)$$

Onde:

- $n$  é o total de *pixels* da imagem;
- $n_q$  é o total de *pixels* que tem intensidade  $r_q$  e
- $L$  é o total de níveis de intensidade na imagem.

O método de Otsu escolhe o limiar de valor  $k$ , tal que  $k$  é um nível de intensidade

que divide o histograma em duas classes  $C_0 = [0, 1, \dots, k-1]$  e  $C_1 = [k, k+1, \dots, L-1]$ , e que maximise a variância  $\sigma_B^2$  definida como:

$$\sigma_B^2 = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 \quad (2.9)$$

Sendo:

$$\omega_0 = \sum_{q=0}^{k-1} p_q(r_q) \quad (2.10a)$$

$$\omega_1 = \sum_{q=k}^{L-1} p_q(r_q) \quad (2.10b)$$

$$\mu_0 = \sum_{q=0}^{k-1} \frac{qp_q(r_q)}{\omega_0} \quad (2.10c)$$

$$\mu_1 = \sum_{q=k}^{L-1} \frac{qp_q(r_q)}{\omega_1} \quad (2.10d)$$

$$\mu_T = \sum_{q=0}^{L-1} qp_q(r_q) \quad (2.10e)$$

O resultado da binarização com limiar ajustado segundo o método de Otsu pode ser observado na Figura 6

IMAGEM AQUI.

Figura 6: Imagem binarizada com limiar definido pelo método de Otsu.

## 2.6 Resumo do processo de extração de características

Como discutido nas seções anteriores deste capítulo, os momentos invariantes foram eleitos como os descritores a serem utilizadas para determinar a similaridade entre as imagens, contudo, estes descritores são extraídos somente após as imagens terem passado por determinadas transformações que visam simplificá-las e potencializar as regiões de maior interesse, e assim, produzir valores mais significativos para os momentos. As transformações aplicadas as imagens são a dessaturação e a binarização, nesta ordem.

Podemos resumir visualmente o processo de extração de característica na Figura 7:



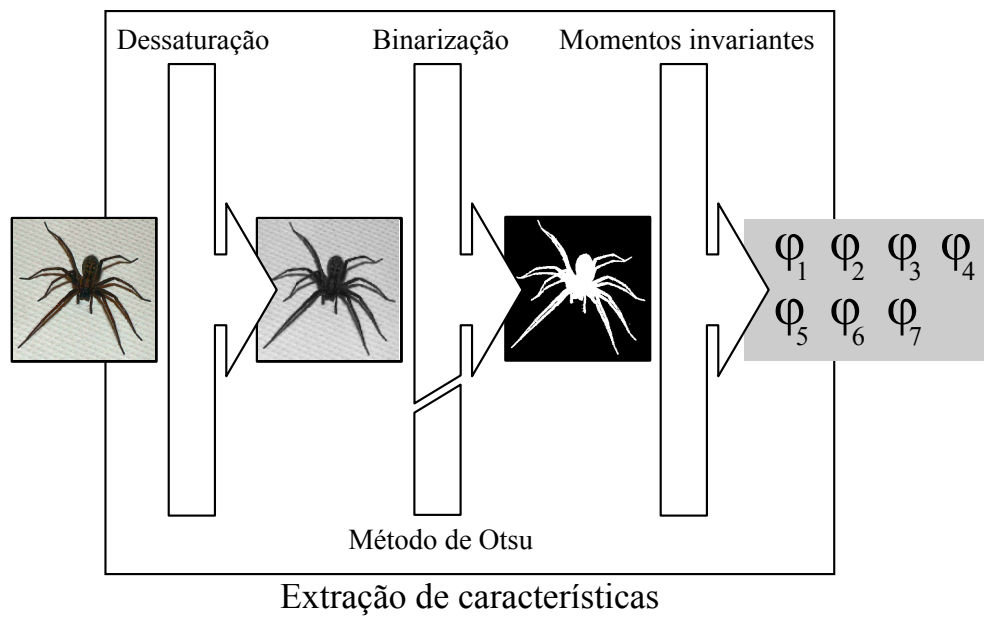


Figura 7: Esquema do processo de extração de características.

## 3 *Rotulação das imagens*

No capítulo anterior foi apresentado o conjunto de características que serão utilizadas para descrever as imagens, e como este conjunto de características é extraído. Neste capítulo o objetivo é demonstrar como este conjunto de características é processado de modo a classificar e agrupar as imagens. As redes neurais de Kohonen são o conceito chave deste processo, contudo, como será visto, um conjunto de técnicas auxiliares também são necessárias para estruturar o *clustering*.

### 3.1 Mapas auto organizáveis

A maneira mais intuitiva de se agrupar imagens, ou qualquer outro tipo de informação, é estabelecer um determinado número de classes e mapear cada imagem para uma das classes, de modo que imagens semelhantes pertençam a mesma classe. Entretanto, este tipo de abordagem desconsidera graus diferentes de semelhança intra-classes, e até mesmo, graus diferentes de semelhança inter-classes, isto é, havendo somente a ligação das imagens com suas classes como será possível determinar, numericamente, o quão semelhante duas imagens são? Ou até mesmo, quão diferentes são duas classes uma da outra?

Estas perguntas são relevantes porque, para os casos onde uma classe possui centenas de imagens, muitas vezes o que se deseja é apenas uma amostra significativa da classe, ou em outra situação, estando em posse de uma determinada imagem, deseja-se um número definido de imagens similares. Por isto, um método voltado exclusivamente para agrupamentos pode não fornecer um conjunto adequado de parâmetros que permitam extrair das classes informações como as necessárias para responder as questões acima.

Partindo de um outro ponto, em vez de iniciar o projeto do *clusterig* pelas classes, mas sim pela distância entre as imagens, surge a possibilidade de criar espaços para posicionar as imagens e, tendo a distância como medida de diferença, definir as classes como regiões ou intervalos dentro destes espaços. Deste modo, as classes não serão um parâmetro para

a classificação, elas serão definidas automaticamente pela dispersão das imagens nestes espaços através de um processo dinâmico e automático. O resultado será não somente um método que agrupa imagens mas que também define, sem a participação ativa do usuário, as próprias classes utilizadas no agrupamento.

Estes espaços onde as imagens serão posicionadas podem ser de qualquer dimensão, contudo, um espaço bidimensional de intervalos discreto é o suficiente para os propósitos deste trabalho. Outro ponto importante é que estes espaços não podem ser infinitos, pois obviamente estão limitados pela memória e pela capacidade de processamento do computador, por isto, serão utilizados sempre espaços limitados. Em resumo, podemos chamar estes subespaços bidimensionais discretos de mapas, onde cada ponto é uma posição potencial para uma imagem, e as imagens estão tão próximas quanto forem semelhantes.

Em resumo, a localização espacial de cada imagem e sua vizinhança topológica representam, em um domínio ou característica particular, nestes casos os momentos invariantes, uma classe, e estas classes são construídas de forma emergente, sem nenhuma comparação com padrões desejados, por isto são ditos auto organizáveis, como indica a Figura 8.

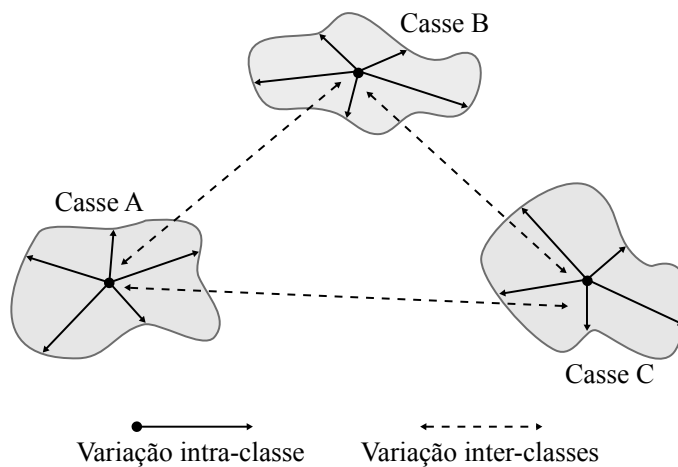


Figura 8: Organização das classes em um mapa auto organizável, coesão interna entre os elementos e isolamento externo entre as classes.

### 3.2 Redes de Kohonen como mapa auto organizável

Dentro da Inteligência artificial, mais especificamente no contexto do aprendizado de máquina, as redes neurais artificiais são sistemas computacionais inspirados na estrutura do cérebro, em particular dos neurônios, que adquirem conhecimento através da experiência.

As redes neurais se assemelham a grafos direcionados, onde os nós são os neurônios, ou unidades de processamento, que possuem uma quantidade indefinida de conexões de entrada e saída. As conexões são o equivalente às arestas do grafo, e são responsáveis por transmitir informações entre os neurônios, podendo amplificar ou reduzir a acuidade destas informações. A Figura 9 apresenta o esquema típico de um neurônio.

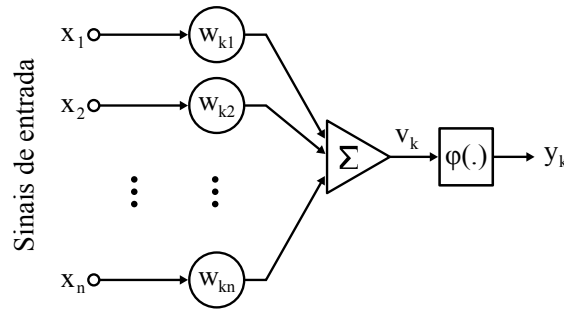


Figura 9: Modelo de neurônio.

Sinais de entrada provenientes de fora da rede chegam por meio de conexões originadas do mundo externo, de modo semelhante, saídas da rede para o mundo externo são conexões que deixam a rede.

A configuração da rede, ou seja, os pesos atuais das conexões, determina como os dados de entrada irão ativar os diferentes neurônios e gerar um determinado resultado. Para grande maioria dos tipos de redes neurais, uma configuração particular é obtida através de um algoritmo de treinamento. O treinamento em geral busca reforçar as conexões que geram bons resultados e penalizar as que não geram.

As redes de Kohonen apresentam apenas duas camadas de neurônios, a camada de entrada e a de saída. A camada de saída é uma espécie de malha de neurônios não conectados entre si, mas amplamente conectados com os neurônios da camada de entrada, como indicado na Figura 10. Esta malha funciona como um mapa, onde para cada padrão de entrada apenas um neurônio é ativado, padrões semelhantes ativam neurônios dentro de uma mesma região da malha.

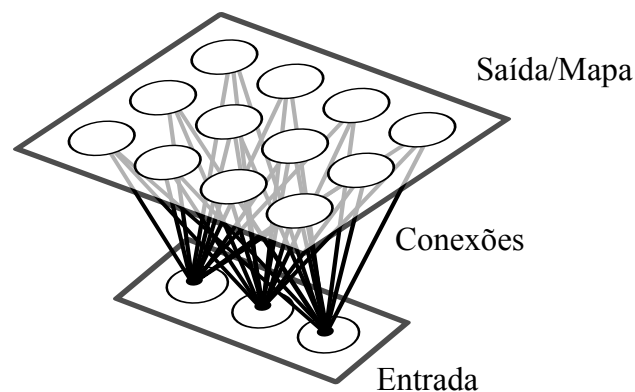


Figura 10: Representação de uma rede de Kohonen.

As redes de Kohonen possuem um algoritmo próprio de treinamento, dividido em três etapas; na primeira, chamado de processo competitivo, uma determinada entrada ativa apenas um neurônio da malha; na segunda, chamado de processo cooperativo, o neurônio escolhido estabelece uma vizinhança de neurônios que serão ajustados para, junto com ele, identificar padrões semelhantes ao que foi apresentado; e por fim, na terceira etapa, chamada de processo adaptativo, os pesos são atualizados com base no neurônio vencedor e na vizinhança topológica. Este algoritmo de treinamento é dito não supervisionado, pois não depende de um par (*entrada, saída esperada*), já que a própria rede estabelece como será a configuração dos resultados, o processo de auto organização da rede.

### 3.3 Características gerais de uma rede neural de Kohonen

Esta seção irá apresentar mais detalhadamente como é a configuração de uma rede de Kohonen e seu algoritmo de treinamento.

#### 3.3.1 Topologia de uma rede de Kohonen

Como dito anteriormente, a rede de Kohonen apresenta apenas duas camadas de neurônios, a camada de entrada e a camada de saída. A camada de entrada deve possuir tantos neurônios quanto forem à quantidade de elementos do padrão de entrada. A camada de saída é uma grade de geometria livre, geralmente retangular, de neurônios que não estão ligados entre si, mas estão, cada um, ligados a todos os neurônios da camada de entrada. As conexões apresentam pesos para escalar o sinal enviado. O esquema conceitual de uma rede de Kohonen é demonstrado na Figura 11:

IMAGEM AQUI.

Figura 11: Esquema detalhado de uma rede de Kohonen.

### 3.3.2 Treinamento da rede

O treinamento requer que os pesos sinápticos sejam iniciados com valores bem pequenos, para que a rede não apresente inicialmente nenhuma configuração. Três processos são executados para cada entrada do conjunto de treinamento, o processo competitivo, o processo cooperativo e o processo adaptativo.

#### Processo competitivo

Quando uma entrada  $x = [x_1, x_2, \dots, x_n]^T$  é apresentado à rede, o neurônio da grade que melhor responder a este padrão será ativado, este neurônio é dito vencedor, e será recompensado ajustando-se seus componentes para mais próximo do vetor de entrada.

O critério escolhido para determinar o neurônio vencedor é a distância euclidiana entre o vetor de entradas e o vetor de pesos das sinapses do neurônio, como indicado na equação 3.1:

$$d_i(t) = \sqrt{\sum_{j=1}^N (x_j(t) - w_{ij}(t))^2} \quad (3.1)$$

Onde:

- $d_i(t)$  é a distância euclidiana entre o vetor de pesos do neurônio  $i$  e o vetor de entradas na iteração  $t$ ;
- $i$  é o índice do neurônio da grade;
- $j$  é o índice do neurônio de entrada;
- $N$  é o número de entradas;
- $x_j(t)$  é o sinal de entrada na entrada  $j$  na iteração  $t$ ;
- $w_{ij}(t)$  é o valor do peso sináptico entre o neurônio de entrada  $j$  e o neurônio da grade  $i$  na iteração  $t$ .

## Processo cooperativo

Estudos biológicos indicam que ao ser excitado, um neurônio estimula seus vizinhos topológicos, de forma que quanto mais próximo um neurônio está do neurônio ativo, mais excitado pelo estímulo do neurônio ativo ele é. O processo cooperativo busca simular este mecanismo biológico.

Em termos matemáticos, o que se deseja é um parâmetro  $h_{ik}$ , dito *vizinhança topológica*, que indica o grau de cooperação entre o neurônio vencedor  $i$  e o seu vizinho  $k$ , que deve ser simétrico em relação ao neurônio  $k$  e deve decrescer constantemente com o aumento da distância  $l_{ik}$ , até que  $\lim_{l_{ik} \rightarrow \infty} h_{ik} = 0$ . A função gaussiana 3.2 atende a estas duas exigências:

$$h_{ik} = e^{\left(-\frac{l_{ik}^2}{2\sigma^2}\right)} \quad (3.2)$$

O parâmetro  $\sigma$  é denominado *largura efetiva da vizinhança*, e deve diminuir a cada iteração, indicando uma tendência de especialização da rede. Neste trabalho o parâmetro  $\sigma$  é a equação 3.3:

$$\sigma(t) = \sigma_0 e^{t/\tau_l} \quad (3.3)$$

Onde:

- $\sigma_0$  é o valor inicial de  $\sigma$ ;
- $t$  é a iteração atual;
- $\tau_l$  é uma constante de tempo.

## Processo adaptativo

O processo adaptativo atualiza os pesos sinápticos a cada iteração, levando em consideração o neurônio vencedor e a vizinhança topológica. O ajuste dos pesos deve decrescer com o tempo, para evitar que novos dados comprometam seriamente o conhecimento já adquirido, substituindo padrões já estabelecidos por novos. Algo semelhante ocorre com o cérebro humano, ao decorrer do envelhecimento o aprendizado vai se tornando mais difícil.

O ajuste  $\Delta w_{ij}$  que a sinapse entre o neurônio de entrada  $i$  e um neurônio da malha  $j$  deve sofrer é expresso pela equação 3.4:

$$\Delta w_{ij} = \eta(t) h_{ik}(t) (x_j - w_{ij}) \quad (3.4)$$

Onde  $h_{ik}(t)$  é o parâmetro vizinhança topológica na iteração  $t$ , referente ao neurônio vencedor  $k$ . O parâmetro *taxa de aprendizagem*  $\eta(t)$  é definido pela expressão 3.5:

$$\eta(t) = \eta_0 e^{t/\tau_l}, \eta_0 \in [0, 1] \quad (3.5)$$

Onde  $\tau_l$  é uma constante de tempo.

### Algoritmo geral de treinamento

O algoritmo 2 resume as três etapas anteriores e descreve todo o processo de treinamento de uma rede de Kohonen:

---

#### Algorithm 2: Treinamento de uma rede de Kohonen

---

**Entrada:**  $\sigma_0$ ,  $\tau_l$ ,  $\eta_0$  e o valor do *erro*

**início**

**repita**

        Calcular a *largura efetiva*  $\sigma(t)$ ;

        Calcular a *vizinhança topológica*  $h$ ;

        Calcular a *taxa de aprendizado*  $\eta(t)$ ;

**para cada conexão faça**

            Calcular  $\Delta w$ ;

            Ajustar o arco;

**fim para cada**

**até** *distâncias euclidianas*  $\leq$  *erro*;

**fim**

---

## 3.4 Normalização dos dados de entrada da rede

Neste ponto já deve estar claro que a entrada da rede será o conjunto dos sete momentos invariantes. Os momentos serão utilizados tanto na etapa de treinamento quanto na classificação das imagens propriamente dita. Contudo, da forma como os momentos são



calculados ainda é necessário que eles passem por uma normalização com o propósito de equalizar os valores dos momentos segundo sua real contribuição para caracterização das imagens, isto é, para cada um dos conjuntos de momentos, sete no total, um ajuste é feito sobre cada valor conforme o gral de variação dos dados, isto porque, se os valores de um conjunto são muito parecidos, pode-se concluir que este conjunto não é uma característica forte para distinguir as diferentes classes de imagens, então deve ter uma influência menor na classificação que os demais com alta variação dos dados.

A normalização também ajusta a faixa dos valores dos momentos. Dado que os valores geralmente são bem pequenos, a normalização amplifica proporcionalmente todos eles e evita problemas envolvendo operações computacionais com números muito próximos de zero.

Sendo  $M$  a matriz com os valores originais dos momentos, na forma:

$$M = \begin{bmatrix} m_{11} & m_{12} & \dots & m_{17} \\ m_{21} & m_{22} & \dots & m_{27} \\ \vdots & \vdots & \ddots & \vdots \\ m_{n1} & m_{n2} & \dots & m_{n7} \end{bmatrix} \quad (3.6)$$

A transformação  $N(M)$  que normaliza todos dos elementos de  $M$  é dada por:

$$N_{ij} = \frac{m_{ij} - \overline{H}_j}{\sigma_j} \quad (3.7)$$

Onde:

- $m_{ij}$  é o momento da linha  $i$  coluna  $j$  de  $M$ ;
- $\overline{H}_j$  é a média de todos os elementos da coluna  $j$  de  $M$ ;
- $\sigma_j$  é desvio padrão  $\sqrt{\frac{\sum_{i=1}^n (m_{ij} - \overline{H}_j)^2}{n-1}}$  dos elementos da coluna  $j$ .

Por fim, cada linha da matriz  $N$  é enviada para rede tanto na etapa do treinamento quanto para classificação da imagem.

## 3.5 A rede neural de Kohonen e o agrupamento das imagens

O conjunto dos pesos sinápticos é o resultado obtido pelo processo de treinamento, eles determinarão a posição de cada imagem no mapa, deste modo, os momentos normalizados de cada imagem devem ser apresentados a rede numa segunda vez para serem classificados.

Contudo, a posição das imagens apenas não indica a que classe pertencem, e na verdade, até agora as classes ainda não foram determinadas, para isso é necessário a adição de outros formalismos que através das posições das imagens e dos pesos sinápticos identifiquem as classes e rotule cada imagem para uma delas.

Nas subseções abaixo serão apresentados dois conceitos que determinam as classes de uma rede de Kohonen, a matriz de distâncias unificadas e a transformada de *watershed*.

### 3.5.1 Matriz de distâncias unificadas

Tendo executado o treinamento da rede de Kohonen, pode parecer que a distância euclidiana entre os elementos mapeados é o único parâmetro para a identificação das classes, contudo, existe também uma distância do vetor de entrada para cada posição do mapa, isto implica que também há uma outra distância entre os neurônios além da distância euclidiana, uma distância que representa o grau de dificuldade de se classificar um elemento, no caso as imagens, em outra posição diferente da que naturalmente lhe seria atribuída. Isto faz muita diferença porque, mesmo tendo as classes que serem formadas por elementos adjacentes, esta não é condição suficiente para identificação dos grupos, é possível que dois elementos próximos pertençam a duas classes distintas, para isto basta que a dificuldade de se classificar um desses elementos na posição do outro seja superior a um limite estipulado.

Fazendo uma análise visual destas considerações, seria como se o mapa possuísse campos de atração, formando vales e picos, sendo os vales as classes e os picos os seus limites, ou fronteiras, como indica a Figura 12.

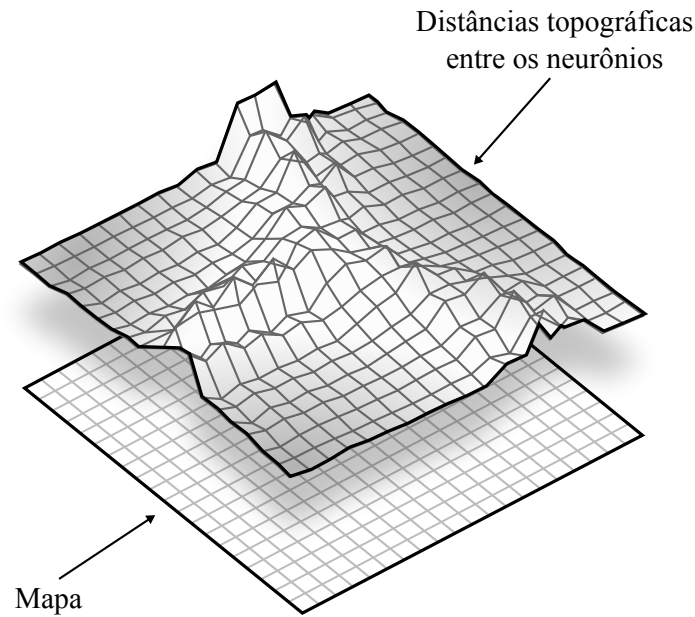


Figura 12: Representação da malha de distâncias topográficas entre os neurônios

O método denominado matriz de distâncias unificadas, ou simplesmente U-Matriz, tem o objetivo de identificar estas relações topológicas, definindo uma função tridimensional onde cada ponto do plano apresenta um valor de distância entre neurônios adjacentes, de modo que valores baixos correspondem a neurônios vizinhos semelhantes e valores altos correspondem a neurônios vizinhos diferentes. Em termos matemáticos, regiões com baixos valores do gradiente, vales, são classes de neurônios especializados em padrões similares e regiões com valores altos correspondem a fronteiras entre as classes.

Considere o mapa retangular discreto limitado de tamanho  $N \times M$ , para cada neurônio  $p$  da camada de saída existe, na U-matriz, três distâncias,  $d_x$ ,  $d_y$  e  $d_{xy}$ , em relação a seus vizinhos, como indicado na Figura 13.

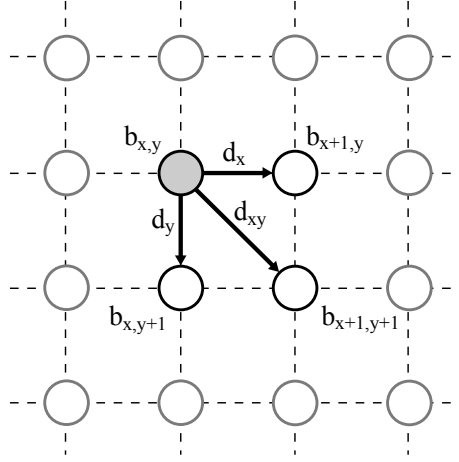


Figura 13: Distâncias  $d_x$ ,  $d_y$  e  $d_{xy}$  entre o neurônio  $b_{x,y}$  e seus vizinhos.

Os valores de  $d_x$ ,  $d_y$  e  $d_{xy}$  são calculados da seguinte maneira:

$$d_x(x, y) = \sqrt{\sum_i (w_{i(x,y)} - w_{i(x+1,y)})^2} \quad (3.8a)$$

$$d_y(x, y) = \sqrt{\sum_i (w_{i(x,y)} - w_{i(x,y+1)})^2} \quad (3.8b)$$

$$d_{xy}(x, y) = \frac{1}{2\sqrt{2}} \sqrt{\sum_i (w_{i(x,y)} - w_{i(x+1,y+1)})^2 + \sum_i (w_{i(x,y+1)} - w_{i(x+1,y)})^2} \quad (3.8c)$$

Estes valores são inseridos em uma matriz de tamanho  $(N - 1) \times (M - 1)$  de acordo com a seguinte tabela:

i	j	$U_{ij}$
$2x + 1$	$2y$	$d_x(x, y)$
$2x$	$2y + 1$	$d_y(x, y)$
$2x + 1$	$2y + 1$	$d_{xy}(x, y)$
$2x$	$2y$	$d_u(x, y)$

Sendo  $c = [c_1, c_2, \dots, c_k]$  o vetor ordenado de elementos circunvizinhos com cardinalidade  $k$ , ainda levando em consideração um mapa retangular, o cálculo de  $d_u$  é obtido pela mediana dos valores circunvizinhos, do seguinte modo:

$$d_u(x, y) = \begin{cases} c_{(k+1)/2}, & \text{se } k \text{ for ímpar} \\ \frac{c_{k/2} + c_{(k+1)/2}}{2}, & \text{se } k \text{ for par} \end{cases} \quad (3.9)$$

Deste modo, a organização da U-Matriz é ilustrada pela Figura 14:

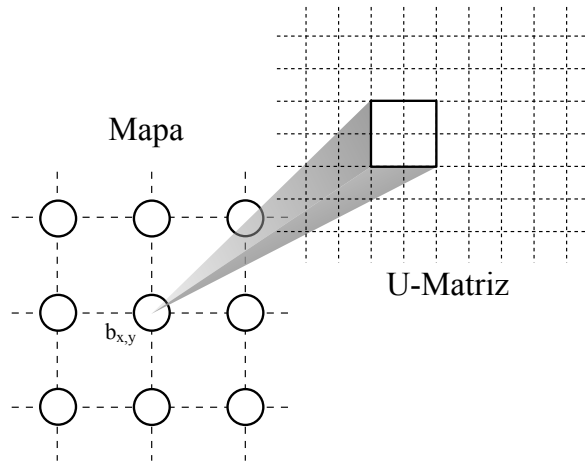


Figura 14: Organização da U-Matriz em relação a grade de neurônios.

### 3.5.2 Transformada *watershed* para rotulação automática da U-Matriz

A transformada de *watershed* estabelece as regiões e fronteiras das classes com base na U-matriz, faz isso apoiada no conceito intuitivo de inundação, vales e diques, da seguinte maneira:

Como visto na Seção 3.5.1, a U-matriz pode ser interpretada como uma superfície que contém vales e montanhas, considerando uma inundação, a água escorre pelas montanhas até os vales, que se inundam com o tempo, formando bacias. Em um determinado momento certas bacias tenderão a se unir, esta união é impedida pela "construção" de diques entre as regiões de fronteira. Ao fim do processo, ou seja, quando a água chegar ao nível da maior montanha, os diques formarão as fronteiras e as bacias formarão as classes. A Figura 15 Ilustra esse processo:

IMAGEM AQUI.

Figura 15: Legenda.

Uma definição formal da transformada de *watershed* passa pela consideração de dois períodos, isto é, pelo processo de expansão das bacias, em outros termos, pela elevação de um nível  $h$  para  $h + 1$ .

Uma bacia associada a um mínimo  $m$  é denominada de  $B(m)$ , os pontos dessa bacia

que possuem altitude menor ou igual a  $h$  são denominados  $B_h(m)$ , isto é:

$$B_h(m) = \{p \in B(m) | f(p) \leq h\} \quad (3.10)$$

O subconjunto de todas as bacias que possuem pontos com altitude menor ou igual a  $h$  é denominado  $X(h)$ , formalmente:

$$X(h) = \bigcup_i B_h(m_i) \quad (3.11)$$

Junto a esses conceitos fundamentais, o conjunto de todas os pontos que pertencem ao mínimo regional  $m_h$  de elevação  $h$  é denominado  $R_{min_h}(f)$ . Esta região será definida posteriormente ainda nesta seção.

Considerando que os primeiros pontos a serem inundados são os pontos mínimos  $h_{min}$ , podemos aplicar a Equação 3.11 da forma:

$$X(h_{min}) = R_{min_{h_{min}}}(f) = T_{h_{min}}(f) \quad (3.12)$$

Onde  $T$  obedece a relação:

$$T_h(f(x)) = \begin{cases} x, & \text{se } h \leq f(x) \\ 0, & \text{qualquer outro} \end{cases} \quad (3.13)$$

Utilizando  $h_{min}$  como ponto de partida, agora é necessário avançar para o estágio onde o nível sobe uma unidade, ou seja, para  $h_{min+1}$ . Neste ponto três situações podem ocorrer, isoladas ou simultaneamente, 1) um novo mínimo será encontrado no ponto  $h_{min+1}$  e formará uma nova bacia, 2) ocorrerá uma expansão da bacia da região cujo mínimo é  $h_{min}$  e 3) duas ou mais bacias distintas de nível  $h_{min}$  estão se expandindo e se encontrarão juntas. Estas três situações são ilustradas na Figura 16:

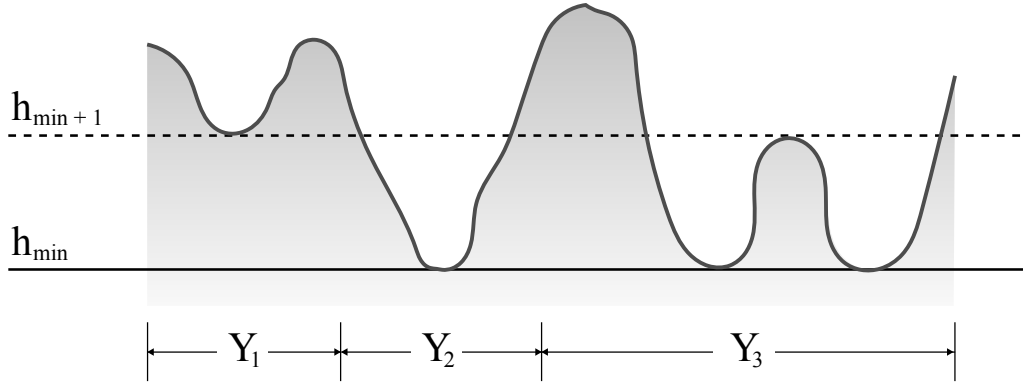


Figura 16: As três possíveis situações na passagem do nível  $h_{min}$  para  $h_{min+1}$ .

1)  $Y_1 \cap X_{h_{min}} = \emptyset$ : Nenhuma bacia foi formada, o que ocorrerá apenas no próximo avanço de nível. Neste caso, vale a relação:

$$\forall p \in Y_1 \begin{cases} p \notin X_{h_{min}} \Rightarrow f(p) \geq h_{min} + 1 \\ p \in Y_1 \Rightarrow f(p) \leq h_{max} \end{cases} \quad (3.14)$$

2)  $Y_2 \cap X_{h_{min}} \neq \emptyset$  e é conectado: Neste caso a inundação já atingiu o mínimo de  $Y_2$  e o processo se encaminha numa expansão da bacia, o que pode ser descrito como:

$$Y_2 = B_{h_{min+1}}(Y_2 \cap X_{h_{min}}) = Z_{Y_2}(Y_2 \cap X_{h_{min}}) \quad (3.15)$$

Onde  $Z_{Y_2}(Y_2 \cap X_{h_{min}})$  é a zona de influência geodésica de  $Y_2 \cap X_{h_{min}}$  contida em  $Y_2$ . Esta zona de influência geodésica  $Z_A(K_i)$  de um componente conectado  $K_i$  dentro de um conjunto  $A$  é o lugar geométrico dos pontos de  $A$  que a distância geodésica para  $K_i$  é a menor que a distância geodésica para qualquer outro ponto de  $A$ , em outros termos:

$$Z_A(K_i) = \{p \in A, \forall j \in [1, N] - \{i\}, d_A(p, K_i) < d_A(p, K_j)\} \quad (3.16)$$

Onde a distância geodésica  $d_A(p, q)$  entre dois pontos  $p$  e  $q$  pertencentes a  $A$  é o menor caminho entre todos os caminhos possíveis de pontos, também pertencentes a  $A$ , que ligam  $p$  e  $q$ .

3)  $Y_3 \cap X_{h_{min}} \neq \emptyset$  e não é conectado: Neste caso  $Y_3$  contém dois ou mais mínimos e eles estão expandindo juntos, denotados de  $M_1, M_2, \dots, M_k$ . Sendo  $M_i$  uma destas regiões a melhor aproximação para  $B_{h_{min+1}}(M_i)$  corresponde a zona de influência geodésica de  $M_i$

dentro de  $Y_3(M_Y)$ :

$$B_{h_{min+1}}(M_i) = Z_Y(M_i) \quad (3.17)$$

Como em 2) e 3) são bacias que estão em expansão, podemos definir estas regiões em termos de uma única zona de influência geodésica  $X_{h_{min}}$ , deste modo  $X_{h_{min+1}}$  é definido como a união destas zonas de influência geodésicas onde os mínimos regionais foram os mais recentemente descobertos, formulado em termos de uma recursão:

$$\begin{cases} X_{h_{min}} = \bigcup_i h_{min_i} \in f \\ X_{h+1} = R_{min_{h+1}}(f) \cup Z_{T_{t \leq h+1}(f)}(X_h) \end{cases} \quad (3.18)$$

Por fim, o conjunto de bacias encontradas na U-matriz representam as classes e servirão para rotular as imagens, para cada mínimo local haverá uma bacia e com isso uma classe. Como o conjunto de mínimos locais pode ser grande, existe então a chance de uma sobre segmentação da U-matriz, por isso é conveniente restringir a quantidade de mínimos locais, no caso, os mínimos locais que geram bacias com regiões muito pequenas, logo, é comum a utilização de filtros gaussianos sobre a U-matriz antes de se aplicar a transformada de *watershed* para regularizá-la.

### 3.6 Resumo do processo de classificação das imagens e do *clustering* como um todo

Agora que foram apresentados todos as teorias utilizadas nesta técnica de clustering, bem como suas devidas justificativas, é conveniente resumir todo o processo de um ponto de vista macro, enumerando e explicando brevemente cada uma das etapas, vejamos:

O dado inicial é o conjunto de imagens que serão classificadas, a primeira etapa consiste em tratá-las e calcular os momentos invariantes de cada uma delas, como descrito no Capítulo 2. O conjunto dos momentos formam a matriz  $M$  e passa pela etapa de normalização como descrito na Seção 3.4. Os momentos normalizados são um a um enviados para rede de Kohonen com o objetivo de treiná-la, este treinamento ocorre como apresentado na Seção 3.3 e é dividido pelas subetapas de competição, cooperação e adaptação, que também são executados em sequência para cada momento individualmente. O envio dos momentos a rede dura o tempo necessário para que o erro seja tolerável, como descrito pelo Algoritmo 2, caso esta tolerância tenha sido atingida antes que todas os momentos



tenham sido processados, a rede é recriada com dimensões maiores, neste caso diz-se que a rede saturou, isto é, atingiu o limite de informação que pode aprender sem ter analisado todos os dados, deste modo a criação de uma nova rede é inevitável e o treinamento deve ser reiniciado. O resultado do treinamento são os pesos sinápticos devidamente ajustados para mapear os momentos, e consecutivamente as imagens. Em seguida dois processos paralelos tem início, o primeiro é a verificação da posição no mapa que cada imagem possui, e o segundo é a definição das classes dentro do mapa através do cálculo da U-matriz e da transformada de *watershed*. Havendo agora as imagens mapeadas e as classes, por fim cada uma das imagens é rotulada com a classe que sua posição está inserida.

A Figura 17 resume visualmente todo o processo:

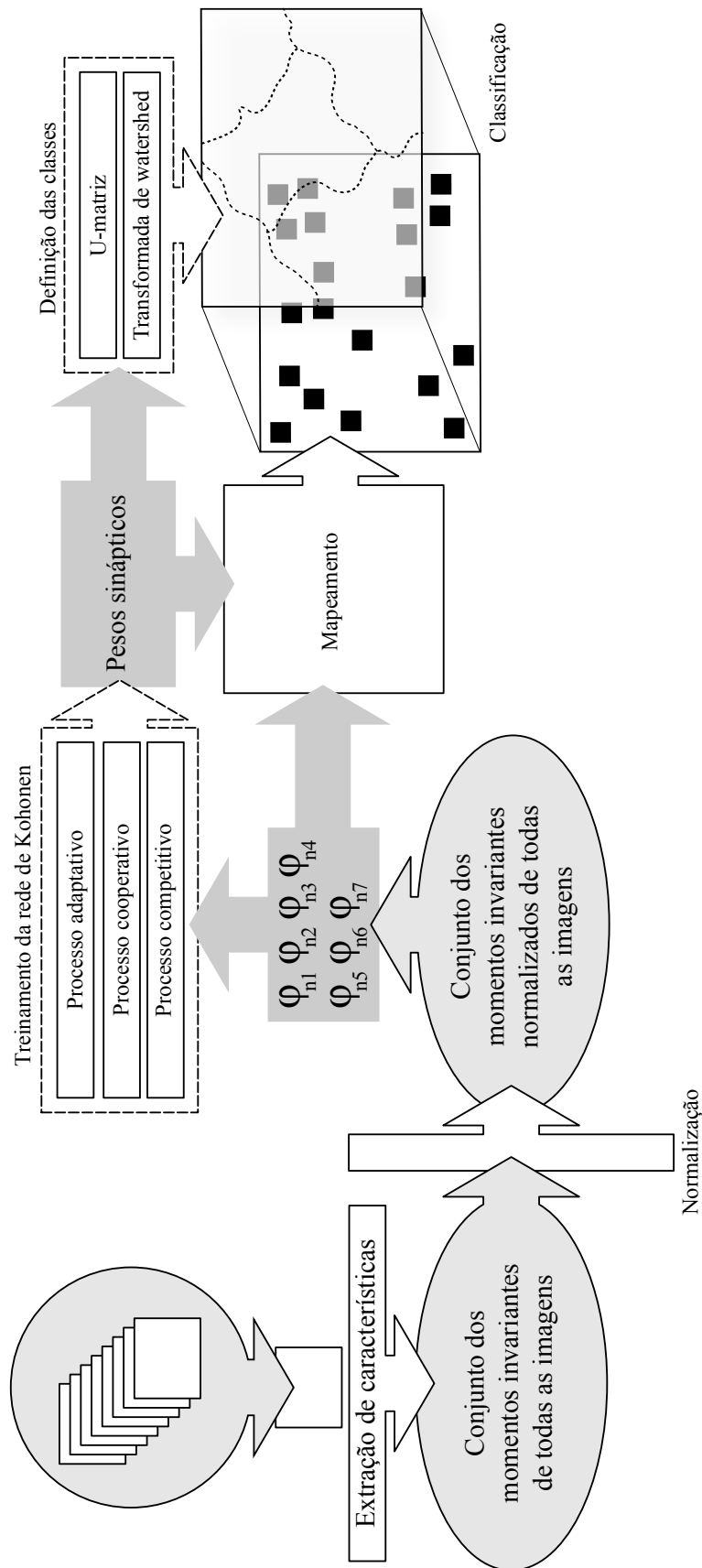


Figura 17: Legenda.

## 4 *Testes empíricos e análise dos resultados*

TEXTO AQUI.

### 4.1 Roteiro e fundamentação do teste proposto

Este trabalho propõe como teste de validação para técnica de classificação descrita nos capítulos anteriores um esquema simples de comparação de expectativas, haverá um grupo de imagens previamente classificado por agentes humanos, e outra classificação, para este mesmo grupo de imagens, gerada pela técnica aqui proposta, deste modo, será possível efetuar a comparação entre a expectativa, isto é, as classes que idealmente devem ser geradas, representadas pela classificação humana das imagens, com as classes fornecidas pela técnica baseada nas redes de Kohonen. As imagens utilizadas no teste são de tal modo que sua classificação humana é auto-evidente, e deste modo, como será observado na Seção 4.2, dificilmente serão alvo de alguma contestação.

O objetivo do teste é avaliar a quantidade e a pertinência das classes, ou seja, se o mesmo número de classes aparece em ambas as classificações e se as classes geradas pelo método proposto possuem correspondência com as classes definidas pelos agentes humanos, mais especificamente, se as imagens estão agrupadas pelo método baseado nas redes de Kohonen do mesmo modo, ou de modo muito semelhante, como foram agrupadas pelos agentes humanos.

### 4.2 Especificação do conjunto de imagens e das classes de controle

O conjunto de imagens utilizada para a execução do teste é o *Columbia Object Image Library* (COIL-100), um conjunto de 100 objetos em 7200 poses diferentes. Destes 100,

foram escolhidas 91 imagens de diferentes objetos para compor o teste. Foram identificadas 15 classes para estas imagens, cada classe possui imagens em poses com variações de translação, rotação e escala para o mesmo tipo de objeto, afim de confrontar as premissas da Seção 2.3 no contexto do processo de classificação.

As classes identificadas foram nomeadas e apresentam as seguintes imagens:

Tabela 1: Grupo A (animais de brinquedo).


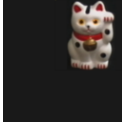


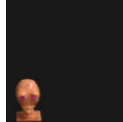





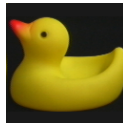
					
obj01.jpg	obj02.jpg	obj03.jpg	obj04.jpg	obj05.jpg	obj06.jpg
					
obj07.jpg	obj08.jpg	obj09.jpg	obj10.jpg	obj11.jpg	

Tabela 2: Grupo B (barquinhos de brinquedo).





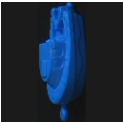

		
obj12.jpg	obj13.jpg	obj14.jpg
		
obj15.jpg	obj16.jpg	obj17.jpg

Tabela 3: Grupo C (boias).




		
obj18.jpg	obj19.jpg	obj20.jpg

Tabela 4: Grupo D (caixas).













						
obj21.jpg	obj22.jpg	obj23.jpg	obj24.jpg	obj25.jpg	obj26.jpg	obj27.jpg
						
obj28.jpg	obj29.jpg	obj30.jpg	obj31.jpg	obj32.jpg	obj33.jpg	

Tabela 5: Grupo E (carrinhos de brinquedo).

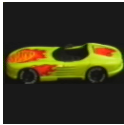
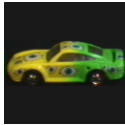
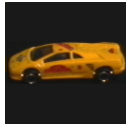
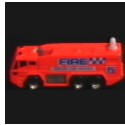
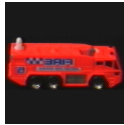
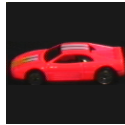




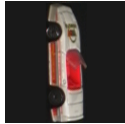

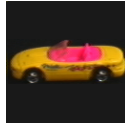

						
obj34.jpg	obj35.jpg	obj36.jpg	obj37.jpg	obj38.jpg	obj39.jpg	obj40.jpg
						
obj41.jpg	obj42.jpg	obj43.jpg	obj44.jpg	obj45.jpg	obj46.jpg	obj47.jpg

Tabela 6: Grupo F (chícaras).

					
obj48.jpg	obj49.jpg	obj50.jpg	obj51.jpg	obj52.jpg	obj53.jpg
					
obj54.jpg	obj55.jpg	obj56.jpg	obj57.jpg	obj58.jpg	obj59.jpg

Tabela 7: Grupo G (embalagens cilíndricas).

					
obj60.jpg	obj61.jpg	obj62.jpg	obj63.jpg	obj64.jpg	obj65.jpg
					
obj66.jpg	obj67.jpg	obj68.jpg	obj69.jpg	obj70.jpg	obj71.jpg

Tabela 8: Grupo H (embalagens retangulares).



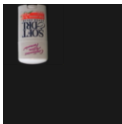





			
obj72.jpg	obj73.jpg	obj74.jpg	obj75.jpg
			
obj76.jpg	obj77.jpg	obj78.jpg	obj79.jpg

Tabela 9: Grupo I (embalagens com tampa).

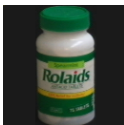
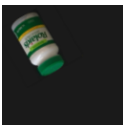









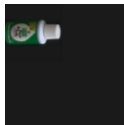
					
obj80.jpg	obj81.jpg	obj82.jpg	obj83.jpg	obj84.jpg	obj85.jpg
					
obj86.jpg	obj87.jpg	obj88.jpg	obj89.jpg	obj90.jpg	obj91.jpg

Tabela 10: Grupo J (ganchos).




		
obj92.jpg	obj93.jpg	obj94.jpg

Tabela 11: Grupo L (lanches).

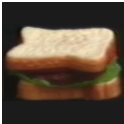


		
obj95.jpg	obj96.jpg	obj97.jpg

Tabela 12: Grupo M (legumes e frutas).


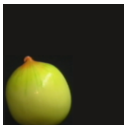
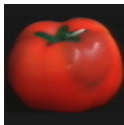

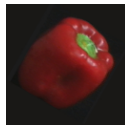






					
obj98.jpg	obj99.jpg	obj100.jpg	obj101.jpg	obj102.jpg	obj103.jpg
					
obj104.jpg	obj105.jpg	obj106.jpg	obj107.jpg	obj108.jpg	

Tabela 13: Grupo N (objetos de madeira).



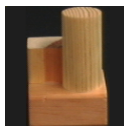
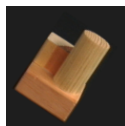

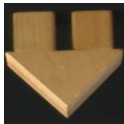


			
obj109.jpg	obj110.jpg	obj111.jpg	obj112.jpg
			
obj113.jpg	obj114.jpg	obj115.jpg	obj116.jpg

Tabela 14: Grupo O (potes).

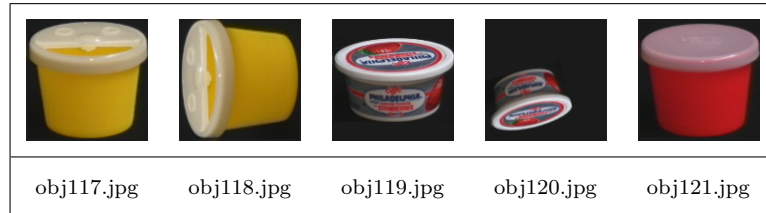


Tabela 15: Grupo P (vasos).



### 4.3 Considerações sobre a implementação e a plataforma de execução

Os testes foram executados em um computador com 64 bits de endereçamento, processador Intel Core i5-2450M CPU 2.50GHz, 2 núcleos e 4 processadores lógicos. O sistema operacional utilizado foi a distribuição Linux Ubuntu 12.04 Precise Pangolin LTS de 64 bits. Os códigos que implementam o método proposto foram todos escritos em ANSI C, e compilados utilizando GCC no modo para gerar código com maior desempenho.

A rede foi configurada para possuir um mapa de  $80 \times 40$  neurônios, o valor inicial para a largura efetiva de vizinhança ( $\sigma_0$ ) foi definido para 0,4, a taxa de aprendizagem inicial ( $\eta_0$ ) para 0,2, e a constante de tempo ( $\tau_l$ ) para 0,2.

### 4.4 Treinamento da rede, convergência do erro e variação da U-matriz

Tendo as imagens, as etapas de (a) binarização, (b) extração dos momentos e (c) normalização foram executadas, as entradas foram organizadas de tal modo que não houvesse



nenhuma sequência grande de imagens pertencente a uma mesma classe, para evitar o enviesamento do aprendizado. Após o treinamento, tendo sido capturado o erro global da rede em cada etapa, a sua convergência ocorreu como demonstrado na Figura 18:

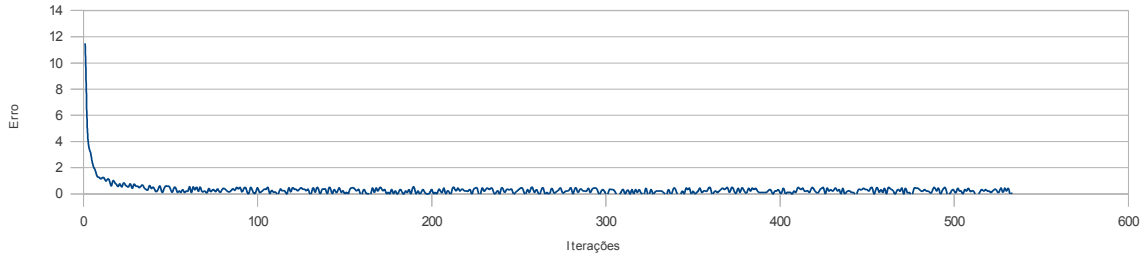


Figura 18: Variação do erro durante as 533 iterações necessárias para completar o treinamento da rede.

A variação da U-matriz em seis momentos específicos pode ser avaliada na Figura 19:

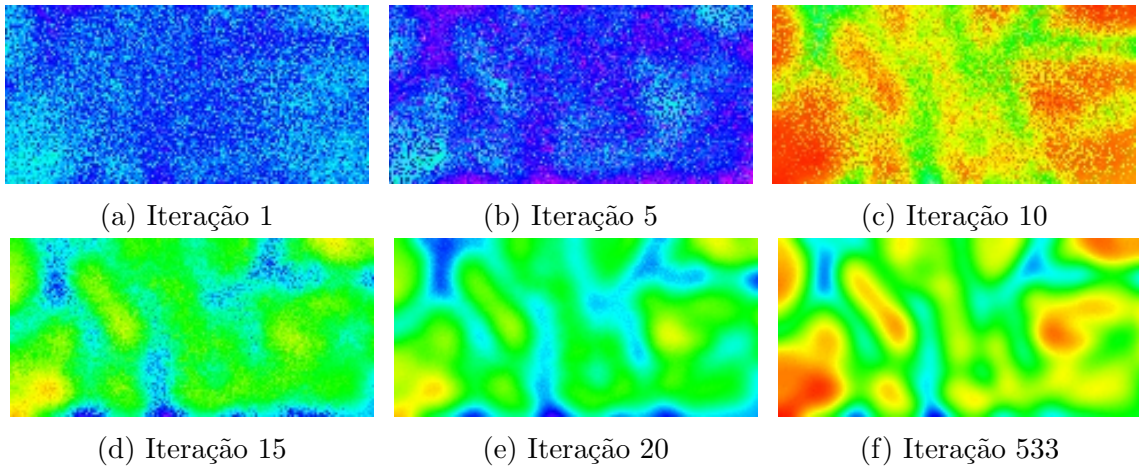


Figura 19: Variação da U-matriz ao longo do treinamento da rede.

Como o erro converge rapidamente para um valor pequeno, e permanesse oscilando não muito acima deste valor até que efetivamente zera. Então as amostras dos estados intermediários da U-matriz foram tiradas de iterações não muito longe do início do processo, e uma do fim para demonstrar o resultado final.

É nítido a presença de vales e fronteiras, e por conseguinte, a presença de grupos identificados.

## 4.5 Tempo de execução

TEXTO AQUI

## 4.6 Disposição das imagens e rotulação das imagens no mapa

A U-matriz resultante do treinamento da rede permite a definição das regiões. As imagens agora podem ser classificadas, no caso, posicionadas na grade e deste modo mapeadas em uma das regiões, concluindo assim todo o processo.

A Figura 20 abaixo representa graficamente os grupos definidos pela transformada de *watershed* sobre a U-matriz resultante:

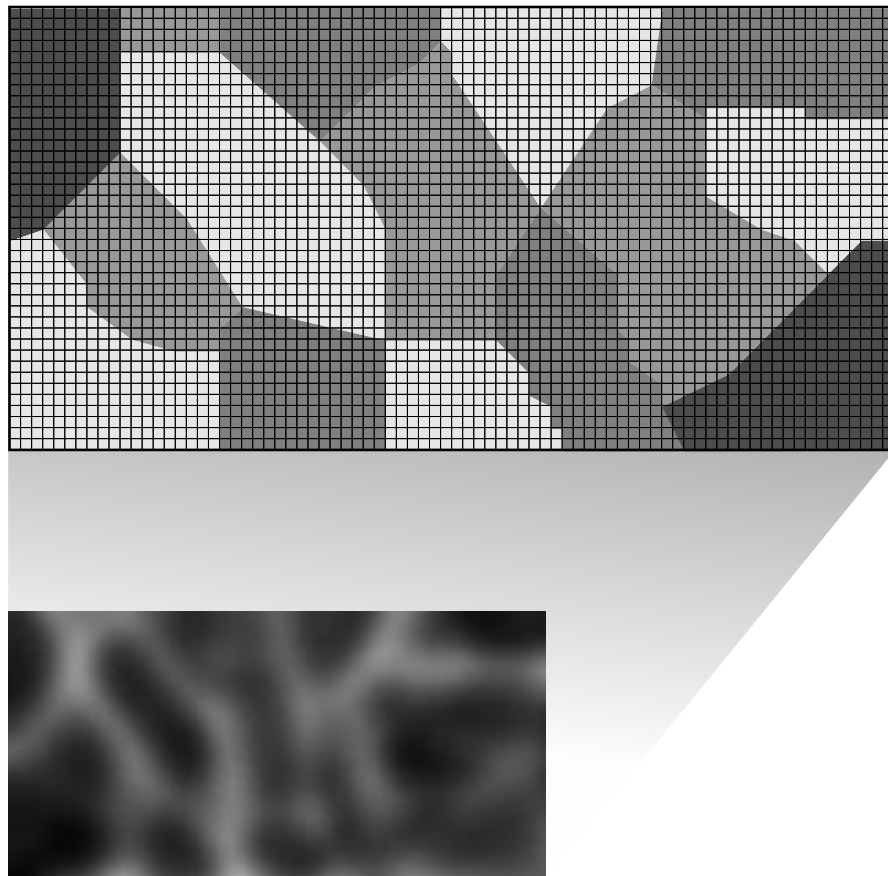


Figura 20: Variação do erro durante as 533 iterações necessárias para completar o treinamento da rede.

Foram 15 grupos identificados no total, cada um deles apresenta as seguintes imagens:

## 4.7 Avaliação geral dos resultados

TEXTO AQUI.

## ***5 Conclusões***

TEXTO AQUI.

## *Referências Bibliográficas*