

# Aplicação do Algoritmo ISAT na Superamostragem de Sinais Bidimensionais

José Alexandre Nalon e Yuzo Iano

**Resumo**—Existem diversos métodos para aumentar a densidade de *pixels* de uma imagem, muitos deles baseados em suas componentes em frequência da imagem, outros baseados em técnicas de inteligência computacional, entre diversos outros. Neste artigo, implementamos um método para realizar a superamostragem de imagens baseado no ISAT (*In Situ Adaptive Tabulation*). Para comprovar a eficiência do método, apresentamos também os resultados de simulações e comparações com métodos clássicos. Ainda que a técnica apresentada seja voltada para imagens, seu uso com sinais de quaisquer dimensões também é possível.

**Palavras-Chave**—Algoritmo ISAT, Interpolação, Processamento de imagens, Superamostragem.

**Abstract**—There are numerous methods to increase pixel density in an image, many of them based in its frequency components, others based in computational intelligence techniques, among others. In this paper, we implement a supersampling method based on ISAT (*In Situ Adaptive Tabulation*). To assess the method's efficiency, we compare the results of our simulations with classical techniques. While the algorithm here is presented with an emphasis on image processing, it could be used with signals with any dimensions.

**Keywords**—ISAT algorithm, Image processing, Interpolation, Supersampling.

## I. INTRODUÇÃO

Uma imagem é um sinal bidimensional  $x[n_1, n_2]$ , geralmente modelado como uma função de duas variáveis discretas  $n_1$  e  $n_2$ , representando respectivamente os eixos horizontal e vertical. Técnicas de processamento de imagens permitem analisar a função em busca de características importantes: quais são essas características depende essencialmente do objetivo do processamento: em sistemas de visão computacional [1], detectar o que são objetos e o que é fundo da imagem; em sistemas de reconhecimento de padrões [2], encontrar, localizar e detectar padrões; em aplicações com objetivo de visualização, a qualidade da imagem é essencial.

Entre os diversos tipos de processamento, o aumento da densidade de pixels da imagem é um procedimento comum, e que deve ser realizado de forma a manter as características da imagem. Logo, deve manter a qualidade visual e o discernimento entre características. Esse processo recebe o nome de *superamostragem* ou *interpolação*, pois consiste em encontrar os valores adequados de *pixels* localizados em posições indicadas por valores fracionários das variáveis  $n_1$  e  $n_2$  — informação que não está presente e só pode ser estimada.

José Alexandre Nalon, Centro Universitário Salesiano de São Paulo (UNISAL), E-mail: jnalon@sj.unisal.br; Yuzo Iano, Faculdade de Engenharia Elétrica e Computação, Universidade Estadual de Campinas (UNICAMP), E-mail: yuzo@decom.fee.unicamp.br.

Esse problema já recebeu diversos tipos de solução. A mais comum consiste em realizar a superamostragem no domínio da frequência. Seja  $X(\omega_1, \omega_2)$  a transformada de Fourier da imagem  $x[n_1, n_2]$ , sendo  $\omega_1 = 2\pi/T_1$  e  $\omega_2 = 2\pi/T_2$ , sendo  $T_1$  e  $T_2$  os intervalos de amostragem na horizontal e vertical, respectivamente.

A superamostragem consiste, efetivamente, em reduzir o intervalo de amostragem por um fator inteiro  $L$ , chamado *fator de reamostragem*, de tal forma que, se  $y[n_1, n_2]$  é a imagem interpolada, então

$$y[n_1, n_2] = \begin{cases} x[n_1/L, n_2/L], & \text{se } n_1/L, n_2/L \text{ inteiros} \\ 0 & \text{caso contrário} \end{cases} \quad (1)$$

Essa operação gera o efeito de contração do espectro, de tal forma que o espectro do sinal resultante é contraído:

$$Y(\omega_1, \omega_2) = X(\omega_1 L, \omega_2 L) \quad (2)$$

Uma vez que o espectro de um sinal discreto é periódico [3] [4], o espectro resultante é filtrado para eliminar componentes de alta frequência. Se  $L = 2$ , o resultado é:

$$Y(\omega_1, \omega_2) = \begin{cases} X(2\omega_1, 2\omega_2), & \text{se } |\omega_1|, |\omega_2| < \pi/2 \\ 0, & \text{caso contrário} \end{cases} \quad (3)$$

Esse procedimento garante que todas as componentes em frequência do sinal original estarão presentes no sinal processado. No entanto, ainda que possa ser matematicamente justificado, não gera as melhores características segundo muitos critérios [5] [6]. O principal efeito notado visualmente é o esmaecimento das bordas e características de alta frequência: a qualidade visual fica prejudicada.

Diversas técnicas surgiram para solucionar esse problema. Em geral, o tratamento dessas características exige algum tipo de processamento não-linear [6]. Um método típico é a filtragem não-linear da imagem, por exemplo, por um filtro de mediana [6]; outras técnicas são realizadas através de redes neurais [7] [8]. Outras técnicas, baseadas em diferentes filosofias, existem, cada uma delas obtendo resultados adequados segundo critérios específicos.

Técnicas de inteligência computacional levam vantagens sobre as clássicas pela sua capacidade de generalização, o que geralmente leva à melhor qualidade visual na imagem resultante. Uma técnica especialmente interessante envolve o uso de redes neurais [8], utilizando o crescimento de uma versão reduzida da imagem original para executar o treinamento. Redes neurais, no entanto, têm o conhecido problema do projeto da arquitetura [9] [10]: é necessário determinar a quantidade de neurônios nas camadas escondidas e um número de outros parâmetros como taxa de aprendizado e fator de esquecimento. A escolha inadequada desses parâmetros pode

fazer com que a rede não atinja um comportamento adequado, podendo ocorrer divergência ou sobreajuste (*overfitting*).

Uma técnica alternativa que não exige a criação de modelos paramétricos é a tabulação das relações entre as variáveis independentes e dependentes. Esse tipo de estratégia recebe o designação “armazenar e recuperar” (*store and retrieve*). O ISAT (*In-Situ Adaptive Tabulation*) [11] [12] é um algoritmo dessa natureza. Criado para aumentar a eficiência da solução de equações diferenciais em reações químicas, é na verdade um método adaptativo para a tabulação de valores de funções. Isso significa que pode ser usado para estabelecer o mapeamento necessário. Basicamente, consiste em um algoritmo que realiza aproximações lineares locais em uma região do domínio de definição da função, associadas a um método para encontrar eficientemente o melhor mapeamento. Adicionalmente, caso a aproximação esteja além de uma margem de erro arbitrária, o algoritmo permite o crescimento da tabulação, de modo a manter a aproximação em níveis adequados.

Como não é um método paramétrico, o algoritmo ISAT encontra algumas vantagens em relação aos métodos baseados, por exemplo, em redes neurais. Entre essas vantagens, citamos a convergência garantida do mapeamento; e a flexibilidade em mapear desde relacionamentos lineares até outros com alto grau de complexibilidade. O algoritmo ISAT já foi usado com sucesso em tarefas como a predição do processo de combustão química [11], controle preditivo [13], controle em tempo real [14] e predição adaptativa [15], entre outros.

Este artigo descreve o algoritmo ISAT e sua utilização para a superamostragem de imagens, e se organiza da seguinte forma: a seção II descreve o algoritmo ISAT; a seção III mostra como utilizá-lo para executar a interpolação das amostras em uma imagem; IV descreve os experimentos realizados e seus resultados, comparando-os com as técnicas mais usuais; a seção V estabelece as conclusões e o direcionamento para pesquisas futuras.

## II. O ALGORITMO ISAT

Esta seção descreve brevemente o funcionamento do algoritmo ISAT. Uma descrição completa e detalhada de seu comportamento pode ser encontrada em [11], [12] e especialmente em [15], em que aplicações em processamento de sinais são apresentadas, e a notação é equivalente à utilizada neste artigo.

O algoritmo ISAT (do inglês *In Situ Adaptive Tabulation*, tabulação adaptativa *in situ*) é um método relativamente recente no conjunto de técnicas capazes de aprender o mapeamento entre diversas variáveis. Foi descrito pela primeira vez em [11], como uma técnica para aumentar a eficiência da solução de equações diferenciais associadas à descrição de processos químicos. Pequenos melhoramentos de desempenho foram apresentados em [12].

Algoritmos de tabulação em geral constroem uma tabela ou outra estrutura de dados equivalente que permite a consulta rápida a valores já calculados da função sendo mapeada. Se aproximações são permitidas, dentro de uma margem de erro  $\epsilon_l$ , definida arbitrariamente conforme as exigências da aplicação, a tabulação então associa uma aproximação adequada a um intervalo na qual é válida. Se uma aproximação

não for possível, então o método deve permitir o crescimento da tabela. Um bom algoritmo dessa natureza permite que isso seja feito de maneira rápida dentro da margem de erro arbitrariamente estabelecida.

O ISAT divide o domínio de definição da função a ser mapeada em diversos subdomínios, nos quais a aproximação é feita de maneira linear. Para determinar em que região da tabulação o vetor inspecionado se encontra, o algoritmo as organiza em uma árvore binária [16], na qual os nós são decisões sobre a posição do vetor no domínio mapeado, e as folhas contém as aproximações. Em um nó, uma região é delimitada por um hiperplano definido por um vetor unitário ortogonal  $\mathbf{n}$  e um ponto  $\mathbf{a}$  a ele pertencente. Ambos os vetores possuem  $N$  componentes, em que  $N$  é o número de variáveis independentes da função sob mapeamento. O ponto investigado estará dentro da região definida pelo hiperplano se [17]:

$$(\mathbf{x} - \mathbf{a}) \cdot \mathbf{n} < 0, \quad (4)$$

em que  $\mathbf{x}$  é o vetor contendo os valores das variáveis independentes sob investigação, e  $\mathbf{x} \cdot \mathbf{n}$  é o produto escalar entre os vetores  $\mathbf{x}$  e  $\mathbf{n}$ . A Figura 1 ilustra o conceito para duas dimensões. O algoritmo percorre a árvore a partir do primeiro nó até encontrar uma folha, que conterá uma aproximação linear, dada pelos parâmetros  $\mathbf{x}_i$ , o centro da região do domínio em que se encontra o vetor inspecionado, e  $\mathbf{A}$ , um vetor-coluna cujas componentes são os coeficientes da aproximação, de forma que a estimativa possa ser calculada como

$$\hat{\mathbf{f}}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_i) + \mathbf{A}(\mathbf{x} - \mathbf{x}_i), \quad (5)$$

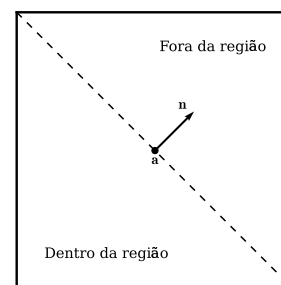


Fig. 1. Determinação da região em que o vetor inspecionado se encontra, em duas dimensões.

A aproximação não será válida, no entanto, em todo o subdomínio na qual é definida. A validade depende da precisão desejada na aproximação, e os próprios parâmetros da linearização permitem determinar em que região essa aproximação é válida. Essa região pode ser determinada através do cálculo do erro quadrático cometido na aproximação, e é dependente dos autovalores e autovetores de  $\mathbf{A}^t \mathbf{A}$  [15]. É uma propriedade conhecida [17] que autovetores e autovalores de uma matriz quadrada podem ser interpretados geometricamente como descrevendo um elipsoide no espaço de  $N$  dimensões. No ISAT, recebe o nome de *elipsoide de precisão* [15].

O algoritmo ISAT pode ser resumido na sequência de passos abaixo:

- 1) Inicializa-se a árvore binária contendo apenas uma região de aproximação, dada por um ponto inspecionado arbitrariamente. Ou seja, a árvore inicial consiste em apenas uma folha. Para essa folha armazena-se o centro  $\mathbf{x}_1$  e os parâmetros da linearização, na forma da matriz  $\mathbf{A}$ , que pode ser calculada por regressão linear ou outro método viável.
- 2) Dado um vetor  $\mathbf{x}$  e uma resposta desejada para esse vetor  $\mathbf{f}(\mathbf{x})$ , determina-se a qual região o vetor pertence. A decisão é feita seguindo o ramo que indica que o vetor está dentro da região caso a Eq. (4) seja satisfeita, e o ramo alternativo caso contrário, até que uma folha da árvore binária seja encontrada.
- 3) Caso o vetor esteja dentro do elipsoide de precisão da região encontrada, retorna-se a aproximação. Caso a precisão não seja atendida, a região é particionada para que uma folha contendo uma nova aproximação seja adicionada à árvore. Isso é feito determinando-se o hiperplano ortogonal ao vetor que une o centro da região ao ponto inspecionado, passando pelo ponto que divide a distância em duas partes iguais.

É importante notar que as estruturas de dados do algoritmo são atualizadas conforme os resultados se fazem necessários. No entanto, em diversos problemas (como a superamostragem, por exemplo), o valor desejado da função não está disponível quando o algoritmo é executado. A seção seguinte descreve como deve ser efetuado o procedimento.

### III. SUPERAMOSTRAGEM DE SINAIS BIDIMENSIONAIS

Algoritmos de tabulação consistem em algoritmos de aprendizagem supervisionada, pois o comportamento do agente modifica-se conforme dados de exemplos, que indicam a resposta desejada para o processamento, lhes são apresentados. Portanto, para que o algoritmo ISAT execute a tarefa de interpolar imagens, é necessária uma fase de treinamento, durante o qual os mapeamentos locais descritos na Seção II são criados e organizados. Esse é um processo comum a todo algoritmo de aprendizado supervisionado [9].

A técnica apresentada aqui é baseada no algoritmo desenvolvido em [8]. Lá, a hipótese feita para a criação dos exemplos do conjunto de treinamento é que a tarefa de realizar a superamostragem pode ser aprendida a partir da interpolação de uma versão reduzida da imagem original. Seja  $x[n_1, n_2]$  a imagem a ser processada, e  $x_R[n_1, n_2]$  uma versão filtrada para remoção de *aliasing* e reduzida por um fator  $L$ . Para simplificar, aqui, assumiremos  $L = 2$ , mas o procedimento pode ser facilmente estendido para trabalhar com quaisquer fatores de reamostragem. Se um método de aprendizagem puder aprender a computar  $x[n_1, n_2]$  a partir de sua versão reduzida, então o mesmo agente pode calcular a versão interpolada a partir da original. Ou seja, assume-se que as tarefas são suficientemente semelhantes de forma que o conhecimento adquirido na primeira pode ser utilizado na realização da segunda.

A Figura 2 mostra o processo de treinamento. Uma imagem é fornecida ao algoritmo é filtrada para remoção de *aliasing*

e reduzida por um fator de 2. Imediatamente, a imagem é reamostrada, e as amostras usadas como resultado desejado. O interpolador, por sua vez, faz uma reamostragem na forma dada pela Eq. (1) para obter os exemplos para o treinamento: o resultado desejado é obtido da imagem original  $x[n_1, n_2]$ , e o vetor de entrada consiste da respectiva amostra e das 8 circundantes na imagem recuperada  $x_R[n_1, n_2]$ . O exemplo é formado pelo par ordenado  $(\mathbf{x}_R[n_1, n_2], x[n_1, n_2])$  em que

$$\mathbf{x}_R[n_1, n_2] = \begin{bmatrix} x_R[n_1 - 1, n_2 - 1] \\ x_R[n_1 - 1, n_2] \\ x_R[n_1 - 1, n_2 + 1] \\ x_R[n_1, n_2 - 1] \\ x_R[n_1, n_2] \\ x_R[n_1, n_2 + 1] \\ x_R[n_1 + 1, n_2 - 1] \\ x_R[n_1 + 1, n_2] \\ x_R[n_1 + 1, n_2 + 1] \end{bmatrix} \quad (6)$$

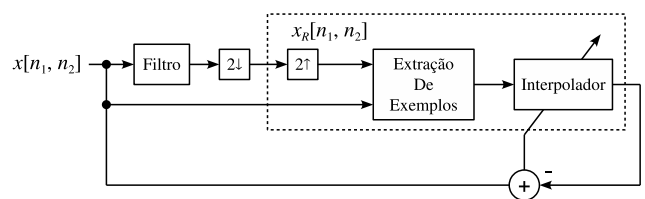


Fig. 2. Processo de treinamento de um algoritmo para a realização da superamostragem de uma imagem.

O conjunto de treinamento é apresentado ao algoritmo até que o treinamento seja considerado completado. Se o interpolador é implementado na forma de uma rede neural, o critério típico para considerar o treinamento completo é o erro quadrático médio. Isso pode envolver a apresentação de algumas épocas, o que depende da topologia utilizada — quanto mais complexa a rede, maior o número de pesos sinápticos e maior a quantidade de exemplos a serem alimentados à rede. O ISAT, por ser um algoritmo de tabulação, não necessita de diversas épocas para que seu treinamento seja considerado completo. Uma vez que toda a imagem tenha sido apresentada, o mapeamento não será melhorado por uma nova apresentação, pois todos os dados passíveis de serem extraídos já o foram. Assim, terminada a apresentação da imagem reconstruída, o treinamento está completado.

A superamostragem da imagem, após o treinamento, é realizada de maneira semelhante, exceto que a resposta desejada não está disponível. A Figura 3 mostra o procedimento: a imagem a ser interpolada é superamostrada como na Eq. (1) e submetida ao interpolador. O vetor de entrada é obtido como na equação (6). A resposta é a imagem desejada.

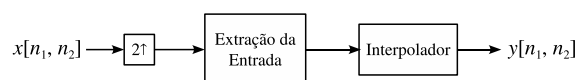


Fig. 3. Realização do processo de superamostragem de uma imagem, após a fase de treinamento.

Essa característica do algoritmo ISAT ressalta uma de suas grandes vantagens sobre algoritmos de aproximações sucessi-

vas — uma vez que a interpolação em uma imagem é um procedimento essencialmente local, e não sendo necessário realizar mais que uma época de treinamento, o interpolador está pronto para executar sua tarefa tão logo os exemplos locais tenham sido apresentados. Em outras palavras, a interpolação pode ser executada *on-line, pixel-a-pixel*. Quando implementado dessa maneira, memória pode ser economizada, e pode-se ganhar tempo de processamento substancial.

#### IV. EXPERIMENTOS E SIMULAÇÕES

Esta seção apresenta o resultado de simulações da superamostragem de imagens utilizando o algoritmo ISAT. Comparamos os resultados com os obtidos pelo uso de algoritmos tradicionais, e especialmente o descrito em [8]. Os resultados são ilustrados com as imagens obtidas, o que permite a inspeção visual. Neste artigo, como medida de distorção, utilizamos a relação sinal-ruído de pico (PSNR). As imagens de distorção são realçadas pelo logaritmo.

A primeira simulação foi realizada com a imagem Lena, com resolução inicial de  $512 \times 512$  pixels. Uma versão reduzida dessa imagem, com resolução de  $256 \times 256$  pixels foi utilizada para treinar os algoritmos envolvidos, segundo o procedimento descrito na Seção III. Com a base de conhecimento gerada, a imagem reduzida foi ampliada até a resolução original. A comparação com o original permite obter a medida de distorção.

A Figura 4 mostra o resultado das simulações. Em 4(a) a imagem original; em 4(b) a imagem ampliada através do crescimento em frequência; em 4(c) a imagem ampliada pelo processamento de uma rede neural e, por fim, em 4(d), o resultado pelo algoritmo ISAT. As relações sinal-ruído de pico para cada um dos métodos foram, respectivamente, 20,1604 dB, 24,1470 dB e 33,9807 dB. O resultado apresentado pelo algoritmo ISAT foi, portanto, significativamente melhor. O erro obtido em cada processamento pode ser visto na Figura 5, realçado pelo logaritmo e em mesma escala: *pixels* mais claro indicam maior magnitude de erro.

Pela análise das figuras, é possível notar que todos os métodos apresentam erro de maior magnitude nas proximidades das bordas dos objetos na imagem. Essa distorção é esperada, pois essas regiões contêm informações de alta frequência que, em geral, não podem ser calculadas, apenas estimadas. O resultado geral é um pequeno esmaecimento dos contornos. Áreas de alta frequência também são afetadas pelo mesmo efeito, o que as torna um bom parâmetro para comparação. Na superamostragem realizada em frequência, efeitos de *aliasing* são visíveis. As redes neurais apresentam bom resultado, mas algum serrilhamento ainda é visível, bem como ineficiência no espaço entre amostras. Esse resultado pode ser efeito da inadequação de diversos parâmetros, como citado anteriormente, que nem sempre são facilmente corrigidos. O algoritmo ISAT mostra boa performance também, com ruído de outra natureza: o ruído encontrado nessas imagens é do tipo impulsivo, nas proximidades das bordas.

Com o objetivo de testar a capacidade de generalização dos algoritmos envolvidos, a base de conhecimento obtida com o treinamento da imagem Lena foi posteriormente aplicada



Fig. 4. Resultado da aplicação de algoritmos de interpolação em uma imagem. Em (a) a imagem original; em (b) ampliada por um método baseado na frequência; em (c) ampliada através de redes neurais e em (d) através do algoritmo ISAT.

à diversas imagens de *benchmarking* comuns em artigos de processamento de imagens, sem que novo treinamento fosse realizado. Em um experimento como esse, espera-se a redução da relação sinal-ruído pois, como se tratam de imagens essencialmente diferentes, o aprendizado realizado para uma imagem não é perfeito para outra. A Figura 6 mostra as imagens do erro cometido para a imagem Baboon. As relações sinal-ruído de pico obtidas foram: 18,4538 dB para a superamostragem em frequência, 19,7996 dB para a realizada pela rede neural, e 19,6120 dB para o algoritmo ISAT. A análise dos resultados mostra que os resultados são comparáveis, embora o algoritmo ISAT tenha se comportado melhor, usando como parâmetro a relação sinal-ruído de pico. Ainda que o desempenho da rede neural tenha sido marginalmente superior para a imagem Baboon, o algoritmo ISAT mostrou considerável ganho para outras imagens. As PSNRs obtidas estão mostradas, em dB, na Tabela I.

TABELA I  
RELAÇÃO SINAL-RUÍDO DE PICO PARA DIVERSAS IMAGENS COMUNS DE BENCHMARKING. OS RESULTADOS SÃO DADOS EM DB.

Imagem	Frequência	Redes Neurais	ISAT
Lena	20,16	24,15	33,98
Baboon	18,45	19,80	19,61
F-16	22,16	24,11	27,88
Boat	25,14	25,23	27,00
Peppers	25,06	22,52	27,30
Pentagon	20,02	18,54	25,47

Os tempos de execução obtidos são da mesma ordem de magnitude para a rede neural e o algoritmo ISAT, embora o primeiro executasse consistentemente mais rápido que o segundo. O método em frequência, como não exige aprendizado,

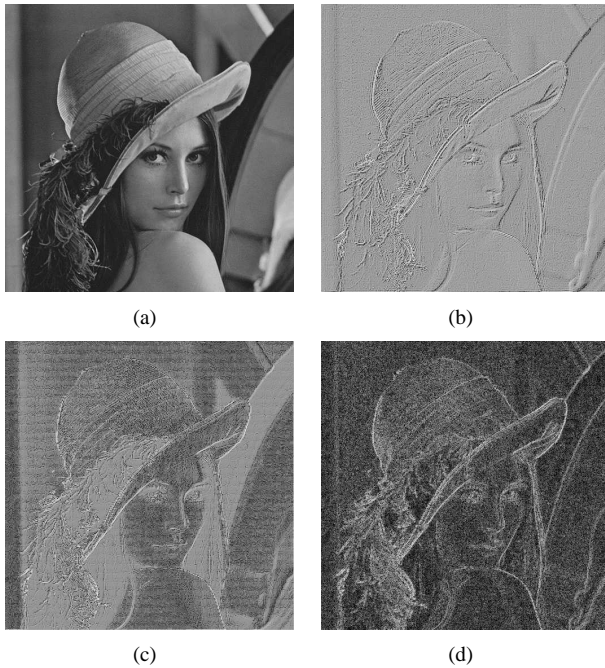


Fig. 5. Erro cometido na interpolação realizada na Figura 4, realçado pelo logaritmo. Em (a), a imagem original, em (b), pelo método baseado na frequência; em (c) através de redes neurais e em (d) através do algoritmo ISAT.

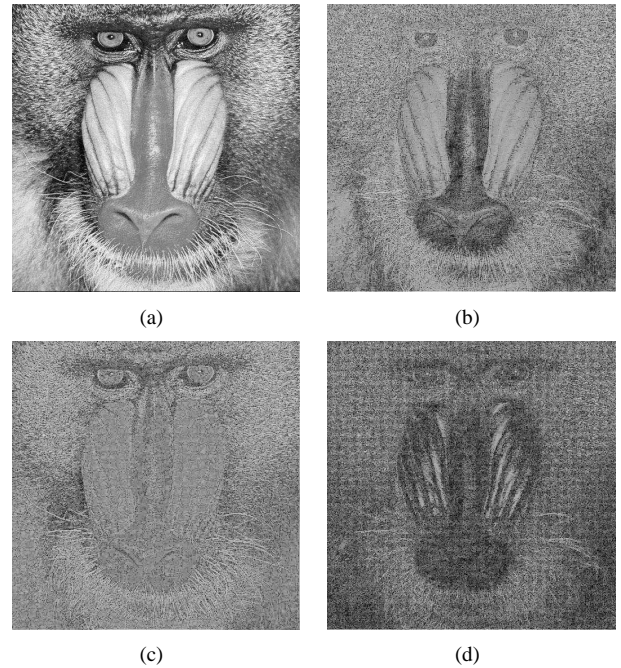


Fig. 6. Erro cometido na interpolação da imagem Baboon com a base de dados gerada pela imagem Lena, realçado pelo logaritmo. Em (a), a imagem original, em (b), o erro cometido pelo método baseado na frequência; em (c) através de redes neurais e em (d) através do algoritmo ISAT.

é consideravelmente mais rápido que os outros dois, mas a qualidade dos resultados desestimula seu uso.

Por ser um algoritmo de tabulação, o ISAT realiza uma ocupação em memória considerável. Na simulação executada, o algoritmo gerou uma árvore contendo 10418 nós e folhas. Esse desempenho é costumeiro em algoritmos dessa natureza, o que os torna inadequados quando recursos de memória são escassos.

## V. CONCLUSÕES

Este artigo descreveu o uso do algoritmo ISAT (*In Situ Adaptive Tabulation*) na interpolação de imagens, tarefa costumeira realizada em sistemas de processamento de imagens. Esse é um algoritmo de tabulação e fornece um método para o rápido armazenamento, recuperação e aproximação de funções de um número arbitrário de variáveis dependentes. A aproximação é linear e realizada localmente, obedecendo uma margem de erro arbitrária.

As simulações realizadas compararam o desempenho do algoritmo sugerido contra outros métodos estabelecidos, e mostraram que ele se comporta consistentemente melhor que esses métodos. Outras simulações compararam a capacidade de generalização do algoritmo, situação em que o ISAT novamente mostrou melhor performance. Os resultados foram analisados visualmente e através da relação sinal-ruído de pico (PSNR), medida costumeira em análise de imagens.

Apesar da aplicação restrita desse artigo, o ISAT é um algoritmo genérico de mapeamento. Pode ser aplicado em qualquer tarefa que exija o aprendizado de uma função qualquer, seja ela linear ou não. Investigações futuras das aplicações do algoritmo em telecomunicações em geral, e processamento de sinais em específico estão planejadas.

## REFERÊNCIAS

- [1] R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2010.
- [2] S. Theodoridis & K. Koutroumbas, *Pattern Recognition & Matlab Intro: Pattern Recognition*, 4th Edition, Academic Press, 2008.
- [3] J. A. Nalon, *Introdução ao Processamento Digital de Sinais*. LTC Editora, 2009.
- [4] A. Oppenheim & R. W. Schaffer, *Discrete-time Signal Processing*, Prentice-Hall, 1989.
- [5] J. S. Lim, *Two-dimensional Signals and Image Processing*, Prentice-Hall, 1989.
- [6] R. C. Gonzales & R. E. Woods, *Digital Image Processing*, 3rd. Edition, Prentice-Hall, 2007.
- [7] N. Plaziac, *Image Interpolation Using Neural Networks*, IEEE Trans. on Image Processing, vol. 8, no. 11, Nov. 1999.
- [8] F. A. C. M. Cardoso, *Algoritmos Genéticos para Interpolação Não-linear de Imagem e Decodificação de Códigos Lineares*, Dissertação de Mestrado, UNICAMP, 1998.
- [9] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed., Prentice-Hall, 1999.
- [10] R. Eberhart & Y. Shi, *Computational Intelligence: Concepts to Implementations*, Morgan Kaufmann, 2007.
- [11] S. B. Pope, *Computationally Efficient Implementation of Combustion Chemistry Using In Situ Adaptive Tabulation*, Combust. Theory Modeling, vol. 1, pp. 41-63, 1997.
- [12] L. Lu & S. B. Pope, *An Improved Algorithm for In Situ Adaptive Tabulation*, Journal of Computational Physics, 228, pp. 361-386, 2009.
- [13] J. D. Hedengren & T. F. Edgar, T. F., *Approximate Nonlinear Model Predictive Control with In Situ Adaptive Tabulation*, Computers and Chemical Engineering, Vol. 32, pp. 706-714, 2008.
- [14] J. D. Hedengren & T. F. Edgar, T. F., *In Situ Adaptive Tabulation for Real-Time Control*, Industrial & Engineering Chemistry Research, Ind. Eng. Chem. Res., Vol. 44, Issue 8, pp. 2716 -2724, 2005.
- [15] J. A. Nalon & Y. Iano, *Predição Não-linear de Amostras com Tabulação Adaptativa In Situ*, XXIX Simpósio Brasileiro de Telecomunicações, 2011.
- [16] A. V. Aho, J. E. Hopcroft & J. D. Ullman, *Data Structures and Algorithms*, Addison-Wesley, 1987.
- [17] H. Anton & C. Rorres, *Álgebra Linear com Aplicações*, Ed. Bookman, 2001.