

Correção de não-uniformidade em vídeo infravermelho por gradiente descendente

Daniel R. Pipa[†], Eduardo A. Barros da Silva[‡], Carla L. Pagliari^{*}

[†]Centro de Pesquisa e Desenvolvimento da Petrobras (CENPES) e UFRJ/COPPE/PEE,
email: danielpipa@gmail.com

[‡] UFRJ/COPPE/PEE, email: eduardo@lps.ufrj.br

^{*} IME, email: carla@ime.eb.br

Resumo—Um problema comum aos sensores infravermelho do tipo IRFPA (*infrared focal plane array*) é o ruído de padrão fixo (*fixed-pattern noise* ou FPN), também chamado de não-uniformidade espacial. De fato, o FPN continua sendo um sério problema apesar dos avanços recentes nessa tecnologia. O presente artigo propõe um método de correção dessas não-uniformidades através de um algoritmo recursivo baseado em gradiente descendente. São utilizadas as diferenças entre dois quadros consecutivos da mesma cena, porém com pequenos deslocamentos, para extrair informações sobre o ruído.

Palavras-Chave—Não-uniformidade, vídeo infravermelho, IRFPA, gradiente descendente.

Abstract—Infrared focal plane array (FPA) sensors suffer from a common problem known as fixed-pattern noise (FPN), or spatial nonuniformity. In fact, FPN remains a serious problem despite recent advances in IRFPA technology. This article proposes a nonuniformity correction recursive algorithm based on gradient descent. The differences between two consecutive frames with small shifts of same scene are used to extract information about the noise.

Keywords—Nonuniformity, infrared video, IRFPA, gradient descent.

I. INTRODUÇÃO

Um problema comum aos sensores infravermelho do tipo IRFPA é o ruído de padrão fixo (*fixed-pattern noise* ou FPN), também chamado de não-uniformidade espacial. De fato, o FPN continua sendo um sério problema apesar dos avanços recentes nessa tecnologia. O presente artigo apresenta o desenvolvimento de um algoritmo de correção dessas não-uniformidades através de um algoritmo recursivo baseado em gradiente descendente. São utilizadas as diferenças entre dois quadros consecutivos da mesma cena, porém com pequenos deslocamentos, para extrair informações sobre o ruído.

O algoritmo proposto neste trabalho adota um modelo simplificado do ruído de padrão fixo onde somente o desvio é considerado. Após definir uma métrica de erro que considera o movimento relativo entre dois quadros e uma estimativa prévio do desvio, é aplicado o gradiente que será usado em equação recursiva para atualizar a estimativa do desvio.

Na seção II é discutida a tecnologia de sensores infravermelhos do tipo IRFPA e seus problemas intrínsecos. Em III são apresentados alguns métodos existentes de correção de não-uniformidade e na seção IV é apresentada uma nova proposta de algoritmo. Na seção V é apresentado o algoritmo de estimação de deslocamentos utilizado. Em VI são apresentados

os resultados de experimentos com vídeos sintéticos e vídeos reais e em VII são feitas as conclusões e considerações finais.

II. INFRARED FOCAL PLANE ARRAYS - IRFPA

Uma *focal plane array* (FPA) é uma matriz de fotodetectores localizada no plano focal de um sistema de lentes, daí seu nome. Tecnicamente, o termo FPA se refere a uma gama de dispositivos sensores de imagem, porém o uso deste termo é mais comum como referência a dispositivos bidimensionais de detectores fotônicos sensíveis ao espectro infravermelho. Ela recebe, então, o nome de IRFPA, ou *infrared focal plane array*. Este trabalho é focado em câmeras de vídeo na região do infravermelho que utilizam a tecnologia IRFPA.

As IRFPA's têm se tornado o mais proeminente detector usado em sistemas de imagens infravermelhas nos últimos anos. O vasto uso desta tecnologia é atribuído aos avanços na tecnologia de sensores de estado sólido, que levaram a compacidade, baixo custo e alto desempenho. Algumas aplicações incluem visão noturna, reconhecimento e vigilância aeroespacial, e imagens térmicas astronômicas [1].

Sabe-se que um problema comum a todos os sensores IRFPA é o ruído de padrão fixo (*fixed-pattern noise* ou FPN), também chamado de não-uniformidade espacial. De fato, o FPN continua sendo um sério problema apesar dos avanços recentes nessa tecnologia. A origem deste ruído é atribuída ao fato de cada detector da matriz, ou seja, cada pixel¹ possuir uma variação no processo de fabricação. Em outras palavras, cada pixel do detector responde de maneira diferente à mesma quantidade de radiação incidente. O FPN se manifesta aleatoriamente no espaço e está presente em todos os quadros de um vídeo infravermelho independentemente da cena ou movimento [1]. Devido ao FPN, a resposta dos pixels da IRFPA em níveis de cinza em relação a incidência de radiação térmica é não-linear.

Outra característica do FPN é sua lenta variação temporal durante o funcionamento do sensor. Ele pode se tornar significativo numa ordem de grandeza de 30 segundos após uma calibração [2]. Esse desvio temporal é atribuído a variações na temperatura do sensor, material de fabricação, ruído eletrônico de leitura, controle automático de ganho, etc. Portanto, uma

¹A palavra pixel não será destacada por já estar contemplada no Dicionário Aurélio da Língua Portuguesa de Novembro de 1999 com o significado: "A menor unidade gráfica de uma imagem matricial, e que só pode assumir uma única cor por vez".

única calibração é ineficaz e o problema requer estimativa e compensação contínuas durante a operação da câmera [1].

Embora a verdadeira resposta das IRFPA's seja não-linear, ela é em geral modelada linearmente como função da radiância, um ganho e um desvio (*bias*) [1], [3] como

$$y_k(i, j) = a(i, j)x_k(i, j) + b(i, j), \quad (1)$$

onde $y_k(i, j)$ é a saída da câmera referente ao pixel (i, j) no instante k , $a(i, j)$ é o ganho associado ao pixel (i, j) , $x_k(i, j)$ é a radiância incidente no elemento sensor (i, j) no instante k e $b(i, j)$ é desvio ou *bias* associado ao pixel (i, j) .

Desta maneira, é possível a correção do FPN através de técnicas que estimem estes parâmetros e apliquem os resultados nas imagens de modo a corrigi-las. Tais correções são normalmente chamadas de NUC (*nonuniformity correction*). O fato do FPN apresentar uma lenta variação faz com que os parâmetros sejam usualmente modelados como invariantes no tempo. Todavia, a variação existe e as correções devem ser aplicadas continuamente.

Em muitos detectores o *bias* é um fator muito mais significativo que o ganho. Assim, bons resultados são obtidos considerando-se apenas o *bias*. Este tipo de modelo contendo apenas *bias* simplifica bastante os algoritmos de correção e também gera bons resultados [4].

III. MÉTODOS EXISTENTES DE NUC

Vários métodos já foram propostos para a correção de não-uniformidade. A seguir são apresentados os principais. Qualquer que seja o método de correção, ele deve ser aplicado repetidamente já que o FPN apresenta uma deriva (*drift*) temporal.

A. Baseados em calibração (Calibration-based)

Trata-se de uma filosofia de calibração onde todos os pixels da IRFPA são expostos a uma temperatura igual de referência, como uma fonte de radiação do tipo corpo negro ou um obturador de temperatura uniforme, de modo a estimar o FPN. Se todos os pixels da IRFPA estão recebendo a mesma quantidade de radiação e deveriam fornecer a mesma saída, o FPN é calculado a partir da diferença das respostas entre os pixels.

O algoritmo mais conhecido de NUC é o de calibração por dois pontos (*two-point calibration*). A idéia é definir um ganho e um desvio (*bias*) para cada pixel de modo a se obter a mesma resposta para toda a matriz quando exposta a mesma quantidade de radiação. A calibração é realizada expondo a IRFPA a dois corpos negros com temperaturas diferentes e conhecidas. Os parâmetros ganho e desvio são obtidos resolvendo-se um conjunto de equações lineares. Após a calibração, a correção da não-uniformidade é obtida subtraindo o desvio relativo e dividindo pelo ganho respectivo a saída de cada pixel da IRFPA, como mostra a equação 2.

$$\hat{x}_k(i, j) = \frac{y_k(i, j) - \hat{b}(i, j)}{\hat{a}(i, j)}. \quad (2)$$

A principal vantagem desse procedimento é sua grande exatidão. Porém, as desvantagens que ele apresenta são:

- Necessidade de recalibrações regulares devido à deriva do FPN.
- Aumento do peso e o tamanho do dispositivo devido à inclusão de corpos negros ou obturadores.
- A visão é interrompida durante o processo de calibração (pode levar alguns segundos).
- Possível introdução de ruído acústico durante a abertura e fechamento do obturador.
- Potência extra para alimentar o obturador.
- A operação normal do dispositivo é interrompida para a realização da calibração.

Se o modelo adotado para o FPN for de ordem maior de modo a melhor representá-lo, mais temperaturas de referências serão necessárias para se estimar os parâmetros do modelo.

B. Baseados na cena (Scene-based)

É uma família de algoritmos que se baseiam na informação contida na própria cena para estimar e corrigir o FPN. Sua grande vantagem é que não há interrupção no vídeo. Eles se dividem em dois tipos: métodos estatísticos (*statistical*) e *registration-based*².

Os métodos estatísticos assumem que todos os pixels são expostos a quantidades de radiação com a mesma distribuição de probabilidade. Esta suposição só é válida se houver variação significativa na cena, movimento da câmera, etc. A correção é obtida ajustando-se os parâmetros de ganho e desvio de cada pixel de modo que suas estatísticas sejam iguais. Se a assunção estatística for violada, o desempenho destes algoritmos tende a ser baixo. Exemplos podem ser encontrados em [5]–[10].

Os algoritmos *registration-based*, por outro lado, necessitam de uma estimativa precisa dos deslocamentos entre quadros consecutivos. Uma vez que esses deslocamentos foram obtidos e os quadros foram alinhados, uma estimativa do FPN é possível comparando-se a resposta de diferentes pixels expostos a um mesmo ponto na cena, ou seja, a mesma quantidade de radiação. A vantagem destes métodos é que não há assunção estatística e a estimativa do FPN e correção são feitas a cada novo quadro. Por outro lado, se a estimativa de deslocamentos contiver erros, não é possível obter uma boa estimativa do FPN. Além disso, os algoritmos tendem a ser mais elaborados e computacionalmente mais custosos. Exemplos destas técnicas podem ser encontrados em [1], [11]–[13]. O algoritmo apresentado neste trabalho é do tipo *registration-based*.

IV. ALGORITMO PROPOSTO

O algoritmo apresentado a seguir é, do conhecimento dos autores, uma nova proposta. Na seção IV-A é apresentado o modelo utilizado e a formulação do problema. Na seção IV-B é apresentado o uso do gradiente descendente e a equação de atualização.

A. Formulação do problema

Seja o modelo simplificado de observação de um sistema de aquisição de imagens infravermelhas com não-uniformidades

²Os autores preferiram manter o termo em inglês por não achar uma tradução adequada ao português.

onde somente o desvio é considerado dado por

$$\mathbf{y}_k = \mathbf{x}_k + \mathbf{b}, \quad (3)$$

onde \mathbf{y}_k é um vetor que representa a imagem observada no instante k , \mathbf{x}_k é um vetor que representa a imagem real no instante k e \mathbf{b} representa o desvio nos dados observados ou *bias*. Todos os vetores que representam imagens estão, neste trabalho, em ordem lexicográfica³. Cada pixel tem seu valor individual de desvio representado por cada elemento do vetor \mathbf{b} . Pelo fato do *bias* possuir uma variação muito lenta, ele é modelado como invariante no tempo [2].

Como o *bias* não é conhecido, pode-se obter uma estimativa da imagem real através de uma estimativa do *bias* através de

$$\hat{\mathbf{x}}_k = \mathbf{y}_k - \hat{\mathbf{b}}. \quad (4)$$

Considerando-se um movimento relativo entre dois quadros consecutivos de uma seqüência de imagens ou vídeo, é possível escrever

$$\mathbf{x}_{k+1} = \mathbf{M}_k \mathbf{x}_k + \gamma_k, \quad (5)$$

onde \mathbf{M}_k é a matriz que implementa o movimento ou deslocamento entre os quadros k e $k+1$, e γ_k é o vetor que modela as inovações no quadro seguinte que não podem ser obtidas com simples deslocamento.

Combinando as equações 4 e 5, desconsiderando o vetor de inovações⁴ γ_k e considerando o deslocamento entre dois quadros como sendo obtido através de um algoritmo de estimativa, é possível escrever uma estimativa para a imagem real do quadro seguinte como

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{M}}_k (\mathbf{y}_k - \hat{\mathbf{b}}). \quad (6)$$

Pode-se ainda combinar as equações 3 e 6 para se obter uma estimativa da próxima imagem observada através de

$$\hat{\mathbf{y}}_{k+1} = \hat{\mathbf{M}}_k (\mathbf{y}_k - \hat{\mathbf{b}}) + \hat{\mathbf{b}}. \quad (7)$$

Pode-se, portanto, definir o erro entre a imagem seguinte observada e a estimativa da imagem seguinte observada com base somente na estimativa de movimento e na estimativa do *bias* através de

$$\epsilon_k = \mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1} \quad (8)$$

$$\epsilon_k = \mathbf{y}_{k+1} - \hat{\mathbf{M}}_k (\mathbf{y}_k - \hat{\mathbf{b}}) - \hat{\mathbf{b}}, \quad (9)$$

onde ϵ_k é o vetor erro de imagem. O erro quadrático médio é definido como

$$\epsilon_k^2 = \frac{1}{N} \sum_{i=1}^N [\epsilon_k(i)]^2. \quad (10)$$

³Entende-se por ordem lexicográfica quando as colunas de uma imagem são empilhadas umas sobre as outras, da primeira à última, para formar um vetor.

⁴O vetor de inovações γ_k foi desconsiderado pois não é possível fazer sua correspondência com nenhuma porção do quadro anterior.

B. Gradiente descendente

Para obter uma estimativa do *bias*, pode-se utilizar o gradiente descendente da seguinte forma:

$$\hat{\mathbf{b}}_{k+1} = \hat{\mathbf{b}}_k - \mu \nabla_{\mathbf{b}} \epsilon_k^2, \quad (11)$$

onde μ representa o passo de atualização, $\nabla_{\mathbf{b}}$ representa o operador gradiente em relação a \mathbf{b} e ϵ_k^2 é o erro quadrático médio como definido anteriormente.

Aplicando o gradiente ao erro, tem-se que

$$\begin{aligned} \nabla_{\mathbf{b}} \epsilon^2 &= \frac{\partial \epsilon^2}{\partial \mathbf{b}} \\ &= \frac{2}{N} \frac{\partial \epsilon}{\partial \mathbf{b}} \epsilon \\ &= \frac{2}{N} \frac{\partial}{\partial \mathbf{b}} (\mathbf{M}\mathbf{b} - \mathbf{b}) \epsilon \\ &= \frac{2}{N} (\mathbf{M}^T - \mathbf{I}) \epsilon. \end{aligned} \quad (12)$$

A equação final de atualização pode ser finalmente escrita como

$$\epsilon_k = \mathbf{y}_{k+1} - \hat{\mathbf{M}}_k (\mathbf{y}_k - \hat{\mathbf{b}}) - \hat{\mathbf{b}} \quad (13)$$

$$\hat{\mathbf{b}}_{k+1} = \hat{\mathbf{b}}_k - \mu (\hat{\mathbf{M}}_k^T - \mathbf{I}) \epsilon_k, \quad (14)$$

onde o termo $\frac{2}{N}$ foi incorporado ao passo de atualização μ . Portanto, a cada novo quadro é feita uma atualização na estimativa do desvio ou *bias* de cada pixel.

V. ESTIMAÇÃO DE DESLOCAMENTO

Foi utilizado o algoritmo de estimação de deslocamento descrito em [1]. O algoritmo foi batizado de LIPSE (*linear interpolation projection-based shift estimator*). Em outras palavras, o algoritmo estima deslocamento baseado na projeção das linhas e colunas e na interpolação através de média ponderada entre elementos vizinhos dos vetores projetados submetidos a diversos deslocamentos inteiros d_i . O fator ponderador (que estima deslocamentos fracionários $0 \leq d_f < 1$) é calculado por fórmula fechada. Abaixo é apresentada sua descrição.

Para cada par de quadros consecutivos fazer:

- Para todos os quadros, calcular os vetores-projeção das linhas \mathbf{y}^l e das colunas \mathbf{y}^c fazendo a média dos pixels de mesma coluna e mesma linha respectivamente. O conjunto desses dois vetores-projeção será chamado daqui para frente de \mathbf{y}^* .
- Para cada par \mathbf{y}_k^* e \mathbf{y}_{k+1}^* , calcular o fator ponderador d_f^* para diversos deslocamentos inteiros d_i^* entre \mathbf{y}_k^* e \mathbf{y}_{k+1}^* .
- Estimar os diversos $\hat{\mathbf{y}}_{k+1}^*$ a partir de \mathbf{y}_k^* e dos deslocamentos inteiros e fracionários $d^* = d_i^* + d_f^*$.
- Calcular o erro quadrático médio (MSE) para os diversos pares \mathbf{y}_{k+1}^* e $\hat{\mathbf{y}}_{k+1}^*$.
- Escolher os deslocamentos na vertical d^l e na horizontal d^c com os menores MSE's.

As vantagens de algoritmos baseados em projeção são rapidez computacional e robustez ao ruído temporal e especialmente FPN, que pode vir a ser um importante impedimento para se alcançar uma estimativa de deslocamento confiável [14]. Para mais informações, consultar [1], [14].

VI. RESULTADOS

Foram realizados experimentos com vídeos sintéticos e infravermelhos reais. Os resultados são apresentados nas seções a seguir.

A. Vídeos com FPN sintético

Foi gerado um vídeo de 50 quadros a partir de deslocamentos circulares aleatórios em uma imagem de tamanho 128×128 pixels. Os deslocamentos foram nas direções vertical e horizontal, assumindo qualquer valor real (parte inteira e fracionária) e com uma distribuição normal de média zero e variância um ($N(0; 2)$).

Foi acrescentado FPN conforme a equação 1. Tanto o ganho quanto o desvio (*gain* e *bias*) foram gerados aleatoriamente com uma distribuição normal de média zero e variância 0,0025 ($N(0; 0,0025)$).

As figuras em 1 (acima e abaixo) mostram respectivamente o quadro número 50 do vídeo sintético antes e depois da correção com o algoritmo desenvolvido. Após estimada a matriz de desvios ou *bias*, a mesma foi subtraída pixel a pixel do quadro.

A figura 2 mostra a evolução do parâmetro SSIM (*Structural Similarity*) de estimação de \hat{x} . O parâmetro SSIM é uma métrica utilizada para comparar a semelhança entre duas imagens [15]. Os valores são apresentados de acordo com o número do quadro e comparando o algoritmo proposto neste trabalho (Pipa) com os algoritmos descritos em [13]. Ambos são baseados em filtro de Kalman e se diferenciam no cálculo da inversa da matriz de covariância do erro. No primeiro caso a inversa é calculada de forma direta. No segundo caso a inversa é calculada através do gradiente conjugado.

B. Vídeos com FPN real

Foi realizada uma filmagem utilizando uma câmera infravermelha marca FLIR SYSTEMS modelo ThermaCAM P65. Seu detector é do tipo *focal plane array uncooled microbolometer*. A resolução utilizada foi 320×240 pixels e foram gravados 100 quadros a uma taxa de 60Hz.

Foi desativada a opção de “Redução do Ruído (*Noise Reduction Off*)” assim como a opção “Período obturação (*Shutter period Off*)”. Esta última diz respeito a correção de FPN realizada pela câmera. Quando esta opção está ativa, a câmera fecha o obturador em períodos de tempo pré-determinados (de 3 minutos a 15 minutos) para efetuar uma calibração como descrito em III-A.

A cena filmada constituía-se de duas seções de tubos de fibra de vidro com defeitos internos introduzidos artificialmente. Atrás dos tubos foi colocada uma fonte de calor, simulando um fluido interno aquecido durante um ensaio não-destrutivo termográfico. Esse tipo de ensaio objetiva detectar reduções de espessura na parede do duto através de diferenças na temperatura da superfície.

A figura 3 mostra na parte superior o quadro número 62 original, e na parte inferior o mesmo quadro após a correção de não-uniformidade utilizando o algoritmo descrito neste trabalho.

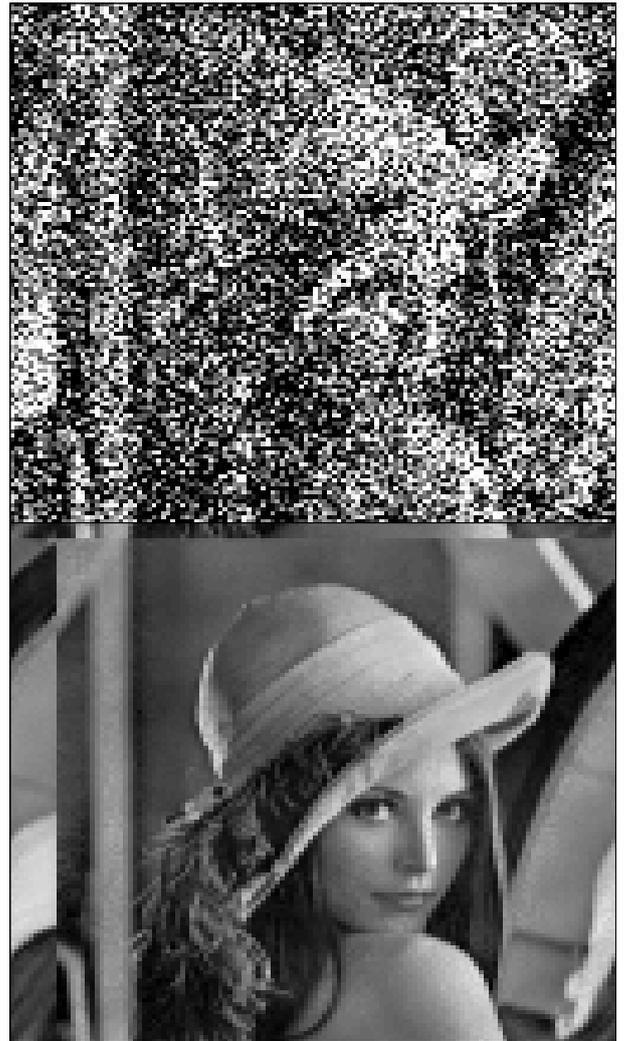


Fig. 1. Vídeo sintético: imagem com FPN (acima) e imagem corrigida (abaixo).

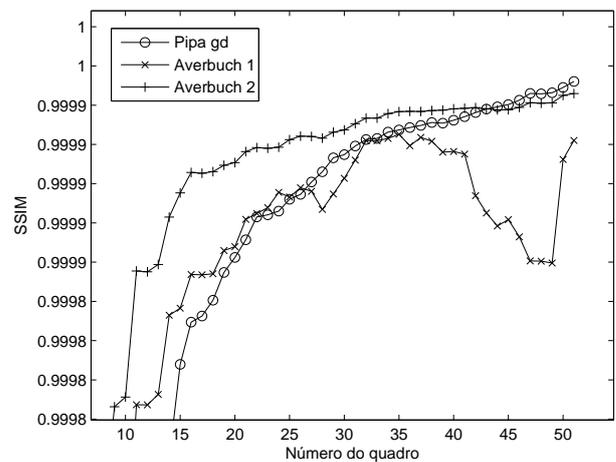


Fig. 2. SSIM de estimação de \hat{x} conforme o quadro.

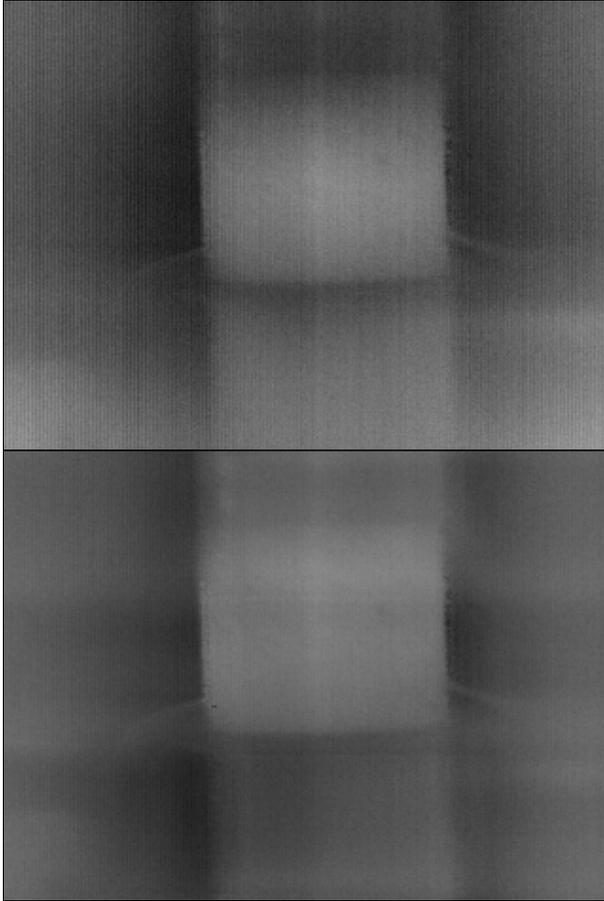


Fig. 3. Vídeo real: imagem com FPN (acima) e imagem corrigida (abaixo).

VII. CONCLUSÕES

Como mostrou a seção VI-A, o algoritmo proposto neste trabalho obteve bons resultados se comparado aos algoritmos propostos em [13]. Além disso, a simplicidade de sua dedução e o rápido desempenho computacional foram fatores superiores e de destaque em comparação com os algoritmos de [13].

Ao se comparar as equações de atualização em [13] com as propostas neste trabalho, percebe-se grande semelhança. O algoritmo proposto em [13] é baseado em filtro de Kalman e, a não ser pela matriz de ganho de Kalman, é equivalente ao deste trabalho. De fato, conseguiu-se chegar a um algoritmo simplificado de [13] com melhor desempenho computacional por uma dedução mais simples.

A título de ilustração, a equação de atualização usada em [13] é dada por

$$\hat{\mathbf{o}}_k = \hat{\mathbf{o}}_{k-1} + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{o}}_{k-1}), \quad (15)$$

onde $\hat{\mathbf{o}}$ é a estimativa do desvio que equivale ao vetor $\hat{\mathbf{b}}$ deste trabalho, $\mathbf{K}_k = \alpha \mathbf{H}_k^T$ é a matriz ganho de Kalman e α é uma constante multiplicativa, $\mathbf{z}_k = \mathbf{y}_k - \mathbf{M}_k \mathbf{y}_{k-1}$ e $\mathbf{H}_k = \mathbf{I} - \mathbf{M}_k$ já fazendo a ponte entre os dois trabalhos. Após alguma manipulação algébrica chega-se a

$$\hat{\mathbf{o}}_k = \hat{\mathbf{o}}_{k-1} + \alpha (\mathbf{I} - \mathbf{M}_k)^T \cdot (\mathbf{y}_k - \mathbf{M}_k (\mathbf{y}_{k-1} - \hat{\mathbf{o}}_{k-1}) - \hat{\mathbf{o}}_{k-1}), \quad (16)$$

que é exatamente a equação 14.

A seção VI-B mostrou resultados interessantes de vídeos reais com FPN. O fato de ter se usado um modelo simplificado de FPN que considera apenas o desvio (*bias*) fez com que estimativa deste variasse mais rapidamente no tempo que o esperado. Embora isso não tenha comprometido os resultados, espera-se que considerando também o ganho seja possível obter resultados melhores.

Os próximos trabalhos incluem a consideração do ganho além do desvio no algoritmo de NUC e a incorporação de um algoritmo de super-resolução [16] (SR) para aproveitar partes do algoritmo de NUC como estimação de deslocamento. Além disso, atualmente as IRFPA's possuem em geral baixa resolução (320×240 ou 640×480) o que justifica um algoritmo de SR.

AGRADECIMENTOS

Os autores, em especial Daniel R. Pipa, agradecem ao Centro de Pesquisa e Desenvolvimento da Petrobras (CENPES) pelo incentivo e permissão da realização do trabalho.

Os autores agradecem também à COPPE/UFRJ, ao IME e o apoio do Centro Tecnológico do Exército-CTEx e da FINEP, sob o convênio no. 2645/06 FINEP/FAPEB/CTEx.

APÊNDICE I DEMONSTRAÇÕES

Segue a seguinte demonstração.

$$\frac{\partial \epsilon^2}{\partial \mathbf{b}} = \frac{\partial}{\partial \mathbf{b}} \frac{1}{N} \sum_{i=1}^N [\epsilon(i)]^2 \quad (17)$$

$$= \frac{1}{N} \sum_{i=1}^N \frac{\partial}{\partial \mathbf{b}} [\epsilon(i)]^2 \quad (18)$$

$$= \frac{2}{N} \sum_{i=1}^N \frac{\partial \epsilon(i)}{\partial \mathbf{b}} \epsilon(i) \quad (19)$$

$$= \frac{2}{N} \sum_{i=1}^N \begin{bmatrix} \frac{\partial \epsilon(i)}{\partial b_1} \\ \frac{\partial \epsilon(i)}{\partial b_2} \\ \vdots \\ \frac{\partial \epsilon(i)}{\partial b_N} \end{bmatrix} \epsilon(i) \quad (20)$$

$$= \frac{2}{N} \begin{bmatrix} \frac{\partial \epsilon_1}{\partial b_1} & \frac{\partial \epsilon_2}{\partial b_1} & \cdots & \frac{\partial \epsilon_N}{\partial b_1} \\ \frac{\partial \epsilon_1}{\partial b_2} & \frac{\partial \epsilon_2}{\partial b_2} & \cdots & \frac{\partial \epsilon_N}{\partial b_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \epsilon_1}{\partial b_N} & \frac{\partial \epsilon_2}{\partial b_N} & \cdots & \frac{\partial \epsilon_N}{\partial b_N} \end{bmatrix} \cdot \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_N \end{bmatrix} \quad (21)$$

$$= \frac{2}{N} \frac{\partial \epsilon}{\partial \mathbf{b}} \epsilon. \quad (22)$$

REFERÊNCIAS

- [1] RATLIFF, B. M. *A Generalized Algebraic Scene-based Nonuniformity Correction Algorithm*. Tese (Doutorado) — The University of New Mexico, 2004.
- [2] SCRIBNER, D. A. et al. Physical limitations to nonuniformity correction in ir focal plane arrays. *Soc. Photo-Opt. Instrum. Eng.*, v. 865, p. 185–202, 1987.
- [3] PERRY, D. L.; DERENIAK, E. L. Linear theory of nonuniformity correction in infrared staring sensors. *Optical Engineering*, v. 32, n. 8, p. 1853–1859, 1993.

- [4] HARDIE, R. C.; DROEGE, D. R. A map estimator for simultaneous superresolution and detector nonuniformity correction. *EURASIP J. Appl. Signal Process.*, Hindawi Publishing Corp., New York, NY, United States, v. 2007, n. 1, p. 206–206, 2007. ISSN 1110-8657.
- [5] HARRIS, J. G.; CHIANG, Y.-M. An analog implementation of the constant average statistics constraint for sensor calibration. In: MOZER, M. C.; JORDAN, M. I.; PETSCHKE, T. (Ed.). *Advances in Neural Information Processing Systems*. The MIT Press, 1997. v. 9, p. 699. Disponível em: <citeseer.ist.psu.edu/article/harris97analog.html>.
- [6] HARRIS, J. *Continuous-time calibration of VLSI sensors for gain and offset variations*. 1995. Disponível em: <citeseer.ist.psu.edu/harris95continuoustime.html>.
- [7] CHIANG, Y.; HARRIS, J. *An analog integrated circuit for continuous-time gain and offset calibration of sensor arrays*. 1997. Disponível em: <citeseer.ist.psu.edu/158320.html>.
- [8] TORRES, S. N.; HAYAT, M. M. Kalman filtering for adaptive nonuniformity correction in infrared focal plane arrays. *Journal of the Optical Society of America*, v. 20, n. 3, p. 470–480, 2003.
- [9] TORRES, J. E. P. S. N.; HAYAT, M. M. Scene-based nonuniformity correction for focal plane arrays using the method of the inverse covariance form. *Applied Optics*, v. 42, n. 29, p. 5872–5881, Oct. 2003.
- [10] SCRIBNER, D. et al. Adaptive retina-like preprocessing for imaging detector arrays. *Neural Networks, 1993., IEEE International Conference on*, p. 1955–1960 vol.3, 1993.
- [11] HARDIE M. M. HAYAT, E. E. A. R. C.; YASUDA, B. J. Scene-based nonuniformity correction with video sequences and registration. *Applied Optics*, v. 39, n. 8, p. 1241–1250, 2000.
- [12] RATLIFF, M. M. H. B. M.; HARDIE, R. C. An algebraic algorithm for nonuniformity correction in focal plane arrays. *Journal of the Optical Society of America*, v. 19, n. 9, p. 1737–1747, 2002.
- [13] AVERBUCH, A.; LIRON, G.; BOBROVSKY, B. Z. Scene based non-uniformity correction in thermal images using kalman filter. *Image Vision Comput.*, Butterworth-Heinemann, Newton, MA, USA, v. 25, n. 6, p. 833–851, 2007. ISSN 0262-8856.
- [14] CAIN, M. M. H. S. C.; ARMSTRONG, E. E. Projection-based image registration in the presence of fixed-pattern noise. *IEEE Transactions on Image Processing*, v. 10, n. 12, p. 1860–1872, 2001.
- [15] WANG ALAN CONRAD BOVIK, H. R. S. Z.; SIMONCELLI, E. P. Image quality assessment: From error visibility to structural similarity. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, v. 13, n. 4, 2004.
- [16] PARK, M. K. P. S.; KANG, M. G. Super-resolution image reconstruction: a technical overview. *IEEE Signal Processing Magazine*, v. 20, n. 3, p. 21–36, 2003.