

Alinhamento de sinais obtidos em trajetórias fechadas utilizando um conjunto genérico de sensores

Allan F. da Silva, Lucas A. Thomaz, Eduardo A. B. da Silva, Sergio L. Netto, Jose F. L. de Oliveira, Alex F. Neves, Guilherme P. S. de Carvalho, Ramon R. Costa PEE, COPPE, Universidade Federal do Rio de Janeiro, Brasil

{allan.freitas, lucas.thomaz, eduardo, sergioln, jose.leite}@smt.ufjr.br

Resumo— Este trabalho apresenta uma nova abordagem para o problema de alinhamento de dados obtidos por sensores em trajetórias similares em momentos distintos. Aqui, sinais são alinhados através de medidas de correlação obtidas por sensores genéricos sincronizados com o sinal de interesse. Os sinais aqui utilizados são gerados a partir de uma trajetória em circuito fechado, de forma que para simplificar o cálculo, é necessário se identificar o período fundamental do sinal e, com ele, segmentar o sinal em uma volta exata. A técnica proposta permite o alinhamento preciso dos sinais, conforme mostram os resultados experimentais obtidos em dados extraídos de ambientes reais.

Palavras-Chave— Alinhamento de Sinais, SLAM, Câmera Móvel, Detecção de Objetos.

Abstract— This article presents a new approach to the problem of aligning data obtained from sensors in similar trajectories in different instants. The signals are aligned using correlation measures obtained from generic sensor ensembles that are synchronized with the signal of interest. The focus signals are generated in a closed-loop trajectory, so it is necessary to identify the fundamental period of the signal to simplify the computation of the correlation. After the obtention of this period the original signal is cropped in a single lap. The proposed technique allows the precise synchronization, as can be seen in the experimental results obtained with data from real-life sceneries.

Keywords— Signals Alignment, SLAM, Moving Camera, Object Detection.

I. INTRODUÇÃO

Em tempos nos quais a captura de sinais (de vídeo, por exemplo) se tornou corriqueira, com sensores disponíveis praticamente em qualquer dispositivo portátil (câmeras portáteis, celulares, vídeo games, etc.) a captura de múltiplos sinais simultaneamente em eventos ou de sinais recorrentes obtidos na mesma localidade se torna cada vez mais comum. A existência deste tipo de conteúdo gera uma demanda crescente do uso de técnicas de alinhamento (espacial e geométrico no caso de vídeos), para que sobreposições temporais possam ser identificadas e conexões de conteúdo possam ser estabelecidas.

Diversos trabalhos na literatura lidam com o problema de alinhamento de sinais, principalmente com aplicações para vídeos. Alguns como [1], [2], [3], buscam alinhar sequências que capturam a mesma cena, isto é, possuem uma parcela considerável de sobreposição visual. Para realizar o alinhamento, alguns dos métodos se utilizam da extração de pontos representativos (*feature points*) de quadros da sequência e realizam um mapeamento entre os pontos obtidos dentro

de sequências distintas, de forma a detectar qual a melhor correspondência espacial e temporal entre os quadros [1]. Outros utilizam análises estatísticas e medidas de similaridade para realizar o alinhamento a partir dos pontos de interesse extraídos [2]. [3] utiliza também informações extras, como sinais de áudio, para realizar o alinhamento.

Um campo menos explorado na área de alinhamento de sequências de vídeo é o alinhamento entre gravações obtidas com câmeras seguindo uma trajetória aproximadamente igual. Em [4], [5] características puramente visuais dos sinais são utilizadas para realizar o alinhamento. No primeiro, histogramas de cores são considerados como medida de similaridade. No outro isso fica a cargo de projeções de homografia.

Um dos primeiros trabalhos a utilizar dados de sincronismo externo com base em sinais de *Global Positioning System* (GPS) foi [6]. O algoritmo apresentado realiza um alinhamento grosseiro a partir do sinal de GPS fornecido e, após esse sincronismo, corrige erros gráficos com a aplicação de uma homografia entre os quadros das sequências de referência e alvo.

Em [7] o alinhamento é realizado através de um método misto em que dados visuais e dados de um sensor de GPS são utilizados para alinhar as sequências de vídeo através de índices. Em outra abordagem mais recente [8], uma estratégia espaço-temporal de minimização iterativa é utilizada para que o alinhamento seja obtido. Características visuais são consideradas para o alinhamento através de descritores visuais e informações obtidas a partir de sinal de GPS.

No campo de métodos de *simultaneous localization and mapping* (SLAM) o alinhamento de sequências de vídeo é bastante utilizado visando a reconstrução de cenários através de câmeras em movimento e scanners a laser. Métodos como [9], baseados em árvores de busca, e [10], que usa álgebra de Lie [11], são aplicados para a realização do alinhamento.

No trabalho que se segue é apresentada uma proposta de alinhamento de sinais obtidos por câmeras que observam recorrentemente a mesma cena. Esta tarefa é realizada com o auxílio de diversos sensores presentes em uma plataforma robótica. Nessa abordagem não é suposto nenhum conhecimento da natureza dos sensores, cujos sinais estão presentes no conjunto disponível para o alinhamento. O algoritmo é utilizado na adaptação do método de detecção de objetos abandonados [5] para outros tipos de trajetória. Muito embora os experimentos realizados aqui sejam de alinhamento de sinais de vídeo, o

método proposto é genérico, de forma que é capaz de realizar o alinhamento de qualquer tipo de sinal.

O restante do trabalho está organizado da forma que se segue: na seção II é apresentada uma visão geral do sistema que gera os dados e dos sensores nele presentes. Na seção III o algoritmo de alinhamento proposto é descrito. Os resultados obtidos são descritos na seção IV e finalmente na seção V são apresentadas as conclusões e discussões sobre o que foi realizado.

II. DESCRIÇÃO DO SISTEMA

O sistema de vigilância consiste em uma plataforma robótica, que pode ser vista na Fig. 1, movendo-se através de um trilho instalado em um locação industrial, realizando uma trajetória fechada com comprimento aproximado de 130 m. A plataforma robótica possui uma câmera em alta resolução que monitora a parte interna de uma planta de processo contendo um ambiente visualmente carregado, além de possuir diversos sensores que medem continuamente variáveis do processo, como temperatura e vibração de equipamentos. A Fig. 2 apresenta um modelo 3D do trilho utilizado. A plataforma robótica, é apresentada em [13], [14], [15]. Nesses trabalhos a aplicação do robô é apresentada e os sensores disponíveis são descritos.

Um modo autônomo de monitoramento utiliza o algoritmo proposto em [5] para a detecção de anomalias no ambiente (como objetos abandonados e a presença de pessoas não autorizadas). Para este fim, alguns procedimentos devem ser adotados. Uma gravação inicial deve ser analisada por um operador, atestando que não há eventos anômalos no ambiente. As gravações subsequentes são comparadas a esse vídeo de referência, de modo a encontrar anomalias.

O trajeto do trilho foi projetado de modo a cobrir regiões de interesse no ambiente e se constitui de um circuito fechado com curvas e retas no espaço tridimensional. Isso torna incompatível com o método de alinhamento presente em [5], que prevê uma trajetória reta com movimento de ida e volta. Uma abordagem simples permite ao sistema prover nativamente uma forma de localização no trilho. A partir da informação de odometria computada através dos encoders dos motores, o sistema estima o deslocamento linear da plataforma móvel durante um intervalo de tempo. Esta estimativa da posição é falha devido ao deslizamento entre a plataforma robótica e o trilho durante o movimento, no entanto isto é o suficiente para alinhar grosseiramente sinais distintos a partir da posição em que o dado foi adquirido.



Fig. 1. Módulo de vigilância instalado em uma planta industrial.

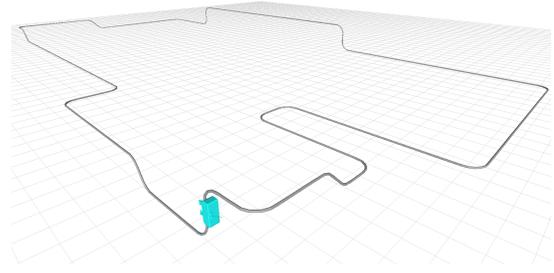


Fig. 2. Modelo 3D do trilho utilizado (em cinza) contendo também a plataforma robótica (em azul).

III. ALINHAMENTO DE SINAIS PARA TRAJETÓRIAS CURVILÍNEAS

O algoritmo proposto neste trabalho visa substituir o alinhamento nativo do sistema por um método de maior robustez, utilizando as informações dos diversos sensores presentes, uma vez que algumas destas, como velocidade angular e linear, são diretamente relacionadas ao movimento realizado ao longo do trilho. Assume-se que durante cada operação o módulo de vigilância mantém uma mesma velocidade média previamente especificada. Com trilho e velocidade média fixos, a cada volta os sinais gerados irão possuir o mesmo padrão de amplitudes, a menos de um atraso referente ao início de cada medição.

O alinhamento entre um vídeo-alvo e uma referência adquirida em uma volta diferente pode ser obtido através dos dados dos sensores calculando-se o atraso relativo $\hat{\delta}$ que maximiza a covariância cruzada entre sinais de referência e alvo de mesma natureza, da seguinte forma:

$$\hat{\delta} = \underset{\delta}{\operatorname{argmax}} \sum_i \sum_n (\mathbf{r}_i(n + \delta) - \bar{\mathbf{r}}_i)(\mathbf{t}_i(n) - \bar{\mathbf{t}}_i), \quad (1)$$

onde $\mathbf{r}_i(n)$ e $\mathbf{t}_i(n)$ representam os sinais adquiridos pelo i -ésimo sensor durante a gravação dos sinais de referência e alvo, e $\bar{\mathbf{r}}_i$ e $\bar{\mathbf{t}}_i$ são as respectivas médias dos vetores \mathbf{r}_i e \mathbf{t}_i .

O cálculo visto na Eq. 1 pode ser feito através da covariância bi-dimensional entre duas imagens. Considerando duas imagens \mathbf{R} e \mathbf{T} , a covariância cruzada é definida como:

$$c_{ij} = \sum_m \sum_n (\mathbf{R}(m + i, n + j) - \bar{\mathbf{R}})(\mathbf{T}(m, n) - \bar{\mathbf{T}}). \quad (2)$$

Agrupando os sinais dos sensores de modo que as i -ésimas linhas das matrizes \mathbf{R} e \mathbf{T} contenham, respectivamente, os sensores \mathbf{r}_i e \mathbf{t}_i , chega-se à seguinte equação para a covariância cruzada:

$$c_{ij} = \sum_m \sum_n (\mathbf{r}_{m+i}(n + j) - \bar{\mathbf{R}})(\mathbf{t}_m(n) - \bar{\mathbf{T}}), \quad (3)$$

Para sinais de média zero, a Eq. 3 se torna idêntica ao argumento da Eq. 1 para um i nulo. Com isso, o processo de otimização assume a forma:

$$\hat{\delta} = \underset{j}{\operatorname{argmax}} c_{0j}. \quad (4)$$

Uma forma computacionalmente eficiente de obter a matriz de correlações é utilizar a DFT bi-dimensional das imagens:

$$\mathbf{C} = [c_{ij}]_{M \times N} = \text{DFT}_{2D}^{-1}[\text{DFT}_{2D}[\mathbf{R}] \circ \text{DFT}_{2D}[\mathbf{T}]^*], \quad (5)$$

sendo \circ a operação de multiplicação ponto-a-ponto.

Note que, para que este cálculo seja efetuado, é necessário que os elementos possuam mesma dimensão, o que pode ser contornado através da técnica de *zero-padding* nos sinais. Em um caso geral, considerando N_1 o comprimento dos vetores \mathbf{r}_i e N_2 o comprimento dos vetores \mathbf{t}_i , os vetores são preenchidos com zeros até possuírem comprimento igual a $N_1 + N_2 - 1$. O Alg. 1 sumariza o procedimento para obtenção do atraso relativo entre o conjunto de vetores.

Algoritmo 1 Alinhamento a partir da máxima covariância cruzada entre dois conjuntos de vetores.

Entrada: Dados de sensores de referência (\mathbf{r}_i) e alvo (\mathbf{t}_i)

Saída: Atraso ($\hat{\delta}$) que maximiza a covariância cruzada entre os dados de referência e alvo

- 1: $N_1 = \text{length}(\mathbf{r}_i)$, $N_2 = \text{length}(\mathbf{t}_i)$
- 2: $\mathbf{r}_i = [\mathbf{r}_i \quad \text{zeros}(1, N_2 - 1)]$
- 3: $\mathbf{t}_i = [\mathbf{t}_i \quad \text{zeros}(1, N_1 - 1)]$
- 4: $\mathbf{R} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_M \end{bmatrix}$, $\mathbf{T} = \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_M \end{bmatrix}$
- 5: $\mathbf{C} = [c_{ij}]_{M \times N} = \text{DFT}_{2D}^{-1}[\text{DFT}_{2D}[\mathbf{R}] \circ \text{DFT}_{2D}[\mathbf{T}]^*]$
- 6: $\hat{\delta} = \underset{j}{\text{argmax}} c_{0j}$

A. Estimaco do perodo do sinal de referncia

O procedimento visto no Alg. 1 para obteno do atraso relativo gera um aumento no tamanho de todos os vetores devido ao clculo da covarincia cruzada atravs da DFT. Entretanto, uma vez que o sistema proposto possui um trilho circular, os sinais obtidos pelos sensores iro possuir naturalmente um comportamento perodico. A obteno do perodo, dado pelo tempo necessrio para o robo percorrer uma volta completa no trilho, permite a menor representao necessria e suficiente do ambiente, e simplifica o clculo da covarincia cruzada.

Para encontrar o perodo, emprega-se a mesma tcnica de covarincia cruzada aplicada nos dados de sensores. Nesta etapa, utilizam-se dados adquiridos em duas voltas consecutivas. Toda a sequncia dos dados  tratada como um conjunto de sinais de referncia perodicos. As amostras iniciais, que pertencem  primeira volta, so copiadas em uma nova varivel e tratadas como sendo um conjunto de sinais de interesse para fins da execuo do algoritmo.

O resultado esperado  encontrar uma covarincia elevada para dois atrasos distintos, indicando a posio de onde os dados foram copiados do conjunto de referncia para o alvo, e as amostras durante a segunda volta que so similares aos dados extrados, que sero aquelas obtidas na mesma posio no trilho. Desse modo, os dois picos na covarincia marcam o tempo necessrio para o robo dar uma volta completa no trilho, ou seja, indica o perodo fundamental dos sinais.

Ao final desta etapa, o algoritmo gera um novo sinal de referncia contendo uma volta completa ao redor do trilho e os dados dos sensores associados. O Alg. 2 sumariza esta etapa.

Algoritmo 2 Obteno do perodo dos dados de referncia.

Entrada: Dados dos sensores de referncia (\mathbf{r}_i) em duas voltas consecutivas, intervalo (N_2) de amostras para usar no clculo da covarincia.

Saída: Perodo (τ), conjunto de vetores (\mathbf{r}_i) e sinal de referncia contendo uma volta exata no trilho

- 1: $N_1 = \text{length}(\mathbf{r}_i)$
- 2: $\mathbf{t}_i = [\mathbf{r}_i(1, \dots, N_2) \quad \text{zeros}(1, N_1 - N_2)]$
- 3: $\mathbf{R} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_M \end{bmatrix}$, $\mathbf{T} = \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_M \end{bmatrix}$
- 4: $\mathbf{C} = [c_{ij}]_{M \times N} = \text{DFT}_{2D}^{-1}[\text{DFT}_{2D}[\mathbf{R}] \circ \text{DFT}_{2D}[\mathbf{T}]^*]$
- 5: $\tau = \text{diferena entre as posio es dos dois maiores picos de } c_{0j}$

B. Alinhamento do sinal alvo

Aps o processamento dos sinais de referncia, a mesma abordagem  aplicada no alinhamento de um sinal alvo oriundo de uma nova gravao. Para esta etapa, no  necessrio que se obtenham dados durante uma volta completa no trilho para o alinhamento dos sinais, o que impossibilitaria uma operao em tempo real, mas sim que haja uma quantidade mnima de dados com informao suficiente para o clculo da covarincia.

Utilizando uma quantidade representativa de amostras dos sinais, os vetores do sinal alvo so preenchidos com zeros at que possuam o comprimento τ dos vetores utilizados na referncia. O algoritmo calcula ento a covarincia cruzada entre os dados de referncia e alvo.

A covarincia resultante deve possuir um pico que indica a posio ao redor da qual as amostras dos dados de referncia so similares s amostras dos dados do sinal alvo, o que vai ocorrer quando os dados comparados forem adquiridos na mesma posio no trilho. Desse modo, o resultado obtido indica o atraso a ser aplicado para que os dados do sinal de referncia registrem um trecho equivalente no sinal alvo, o que implica no alinhamento temporal dos sinais. O Alg. 3 descreve o procedimento para obteno do atraso entre os vdeos de referncia e alvo.

Algoritmo 3 Alinhamento a partir da mxima covarincia cruzada de forma eficiente.

Entrada: Dados de referncia (\mathbf{r}_i) contendo um nico perodo e dados dos sinais alvo (\mathbf{t}_i)

Saída: Atraso ($\hat{\delta}$) que maximiza a covarincia cruzada entre os dados de referncia e alvo

- 1: $N_1 = \text{length}(\mathbf{r}_i)$, $N_2 = \text{length}(\mathbf{t}_i)$
- 2: $\mathbf{t}_i = [\mathbf{t}_i \quad \text{zeros}(1, N_1 - N_2)]$
- 3: $\mathbf{R} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_M \end{bmatrix}$, $\mathbf{T} = \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_M \end{bmatrix}$
- 4: $\mathbf{C} = [c_{ij}]_{M \times N} = \text{DFT}_{2D}^{-1}[\text{DFT}_{2D}[\mathbf{R}] \circ \text{DFT}_{2D}[\mathbf{T}]^*]$
- 5: $\hat{\delta} = \underset{j}{\text{argmax}} c_{0j}$

TABELA I

COMPARAÇÃO ENTRE O PERÍODO OBTIDO PELO ALGORITMO E PELA ESTIMATIVA DE POSIÇÃO LINEAR. CONFIGURAÇÃO 1: AMOSTRAGEM DE 25 HZ E VELOCIDADE MÉDIA DE 0,1 M/S. CONFIGURAÇÃO 2: AMOSTRAGEM DE 10 HZ E VELOCIDADE MÉDIA DE 0,2 M/S.

	Período (número de amostras)	
	Algoritmo	Estimativa da posição
Configuração 1	30985	30947
Configuração 2	6042	6033

IV. RESULTADOS

No experimento realizado os sinais de interesse a serem alinhados são vídeos capturados por uma câmera na plataforma robótica, embora o método desenvolvido não seja especificamente desenhado para o alinhamento de sinais de vídeo e possa ser aplicado a sinais genéricos. A câmera utilizada foi configurada de modo a registrar vídeos com taxa de 25 Hz e resolução de 800×450 . Diversos sinais foram adquiridos no ambiente industrial descrito, com a plataforma robótica se movendo a uma velocidade média de 0,1 m/s, e transmitindo os sinais a um operador via *wi-fi*.

Foram realizados testes da obtenção do período dos sinais de referência e de um novo conjunto de sinais. Os algoritmos foram implementados em Matlab [12] e C++.

Os resultados foram comparados aos valores obtidos pela estimativa do deslocamento linear nativa do sistema. Esta estimativa permite uma forma de obtenção do período, como o tempo necessário para ser percorrido o comprimento total do trilho (130 m), e também uma forma de alinhamento, buscando, para cada amostra do vídeo alvo, a amostra do vídeo de referência adquirida na posição do trilho mais próxima.

A. Obtenção do período da referência

O algoritmo foi testado na obtenção do período do vídeo de referência, a partir de um conjunto de dados contendo duas voltas consecutivas. A Fig. 3 apresenta um exemplo da obtenção do período. Nesta figura, um trecho do sinal (a) é copiado para uma estrutura, gerando o sinal (b). A correlação entre os conjuntos de sinais gera o sinal (c), cujos picos indicam o mesmo lugar no trilho. Este cálculo possibilita a estimativa do tempo em que se efetua uma volta completa, através do cálculo da distância entre as posições dos picos.

A Tab. I apresenta o período obtido para o conjunto de sinais de referência presente na base de dados. Este valor é comparado ao obtido a partir do sistema de localização presente no robô. O método também é testado com uma nova gravação contendo sinais adquiridos a uma taxa de amostragem de 10 Hz e com o robô se movendo a uma velocidade de 0,2 m/s. A Fig. 3 (d) apresenta um trecho do sinal contido na Fig. 3 (a) contendo um período completo.

B. Alinhamento dos sinais de referência e alvo

O teste do alinhamento foi realizado utilizando os sinais de referência obtidos pelo algoritmo na etapa anterior. A Fig. 3 também apresenta um exemplo de cálculo do atraso necessário para o alinhamento entre sinais de referência e alvo. O sinal (d) contém 8 mil amostras adquiridas em uma gravação posterior. A covariância cruzada entre o conjunto de sinais de referência

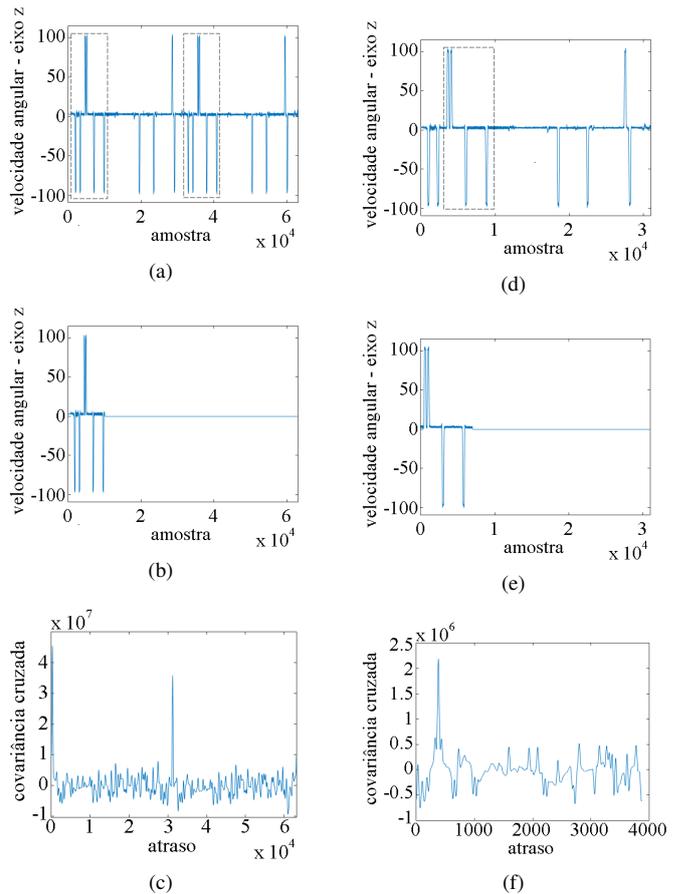


Fig. 3. Exemplo de estimação do período e do atraso relativo entre os sinais. As linhas tracejadas indicam as prováveis regiões do sinal de referência com maior similaridade ao sinal alvo. (a) Sinal de referência de um sensor contido na base de dados durante duas voltas. (b) Amostras do sinal (a) utilizadas como sinal alvo. (c) Covariância cruzada do conjunto de sensores para obtenção do período da referência. (d) Período completo do sinal de referência (a). (e) Amostras contidas na base de dados de sinais alvo do mesmo sensor. (f) Covariância cruzada do conjunto de sensores para obtenção do atraso relativo entre os sinais.

e alvo, contida na imagem (f), apresenta o atraso que maximiza a similaridade entre as amostras dos sinais, ou seja, o atraso necessário para realizar o alinhamento.

Foram realizados testes variando o número de amostras dos sinais alvo. A Fig. 4 apresenta a covariância cruzada obtida para dois conjuntos de sinais, obtidos em posições distintas e com três diferentes quantidades de amostras.

Os resultados comprovam que a quantidade de amostras utilizadas nos sinais alvo possui impacto no resultado do algoritmo. De posse de um número pequeno de amostras, pode não haver informação suficiente para que o algoritmo encontre uma única região com grande similaridade entre os sinais de referência e alvo. Este fato ocorre no sinal utilizado no cálculo da imagem (d), cuja gravação foi iniciada um trecho reto do trilho. Com poucas amostras, o algoritmo não é capaz de distinguir em qual das regiões retas do trilho os sinais foram adquiridos. Já a gravação utilizada para o cálculo da imagem (a) foi iniciada ao lado de uma curva em um formato bem característico, de modo que mesmo com poucas amostras existem informações únicas que permitem diferenciar esta região de todas as outras.

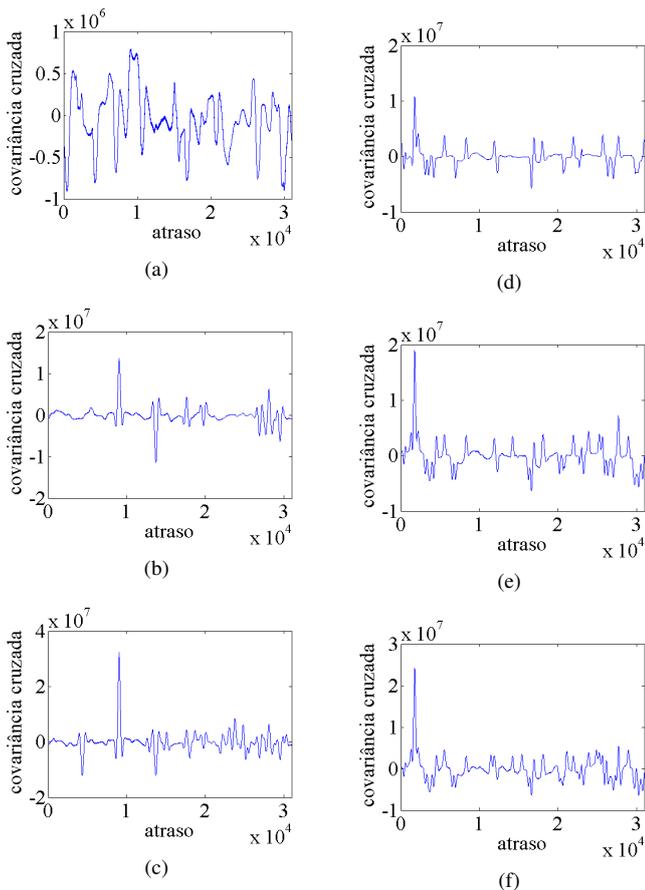


Fig. 4. Covariância cruzada utilizando conjuntos de sinais alvo adquiridos em diversas posições no trilho. (a) Sinal com 2000 amostras iniciado em uma região reta. (b) Sinal com 5000 amostras iniciado em uma região reta. (c) Sinal com 10000 amostras iniciado em uma região reta. (d) Sinal com 2000 amostras iniciado em uma região curva. (e) Sinal com 5000 amostras iniciado em uma região curva. (f) Sinal com 10000 amostras iniciado em uma região curva.

A Tab. II compara o alinhamento obtido pelo método proposto com o alinhamento a partir da estimativa do deslocamento linear, apresentando o erro cometido, em número de quadros, ao substituir um alinhamento pelo outro. Nota-se que, com pelo menos 5000 amostras, que correspondem a um conjunto de dados gravado durante 1/6 da trajetória total do trilho, o erro é aproximadamente 2 quadros, que é por volta de 1 cm da posição linear. Uma vez que o alinhamento será utilizado no algoritmo proposto em [5], um erro dessa magnitude não deve influenciar no resultado, já que existe uma etapa de registro espacial capaz de compensar pequenos deslocamentos.

V. CONCLUSÕES

Este trabalho apresentou um método para realização do alinhamento de sequências de vídeo sincronizadas a um conjunto de sensores presentes em uma plataforma robótica movendo-se ao longo de um trilho. Através da maximização da similaridade dos sinais obtidos, obtém-se o atraso que permite o melhor alinhamento entre dois sinais de vídeo. O método foi testado e comparado a uma odometria presente no sistema robótico, que estima o deslocamento linear no trilho. Os resultados

TABELA II

ERRO EM NÚMERO DE QUADROS ENTRE O ALINHAMENTO FORNECIDO PELO ALGORITMO PROPOSTO E A ESTIMATIVA PELO DESLOCAMENTO LINEAR PRESENTE NO SISTEMA, PARA DIVERSAS QUANTIDADES DE AMOSTRAS DOS SINAIS ALVO EM TRECHOS DIFERENTES DO TRILHO. GRAVAÇÃO 1: INICIADA EM UMA REGIÃO RETA DO TRILHO. GRAVAÇÃO 2: INICIADA EM UMA REGIÃO CURVA DO TRILHO

Tamanho dos sinais	Gravação 1		Gravação 2	
	Média	Variância	Média	Variância
2000	57,95	0,57	3,20	2,83
5000	2,64	0,90	1,99	1,59
10000	2,33	0,93	2,20	2,83
15000	1,88	1,44	1,03	3,51

comprovam que para um conjunto com uma quantidade suficiente de amostras, o método possui comportamento similar ao alinhamento nativo do sistema, com desempenho suficiente para a aplicação considerada.

AGRADECIMENTOS

Este trabalho foi parcialmente financiado por Statoil Brazil Oil & Gas Ltda e Petrobras S.A. sob o contrato COPPETEC 0050.0079406.12.9 com a ANP (Projeto DORIS) e pelo CNPq.

REFERÊNCIAS

- [1] Caspi, Y. e Irani, M., "Spatio-Temporal Alignment of Sequences". *IEEE Transaction on Pattern Analysis and Machine Intelligence*, v. 24, no. 11, pp. 1409–1424, Novembro 2002.
- [2] Douze, M., Revaud, J., Verbeek, J., Jégou, H. e Schmid, C., "Circulant Temporal Encoding for Video Retrieval and Temporal Alignment". *International Journal on Computer Vision*, pp.1–16, Dezembro 2015.
- [3] Dai, C., Zheng, Y. e Li, X., "Accurate Video Alignment Using Phase Correlation". *IEEE Signal Processing Letters*, v. 13, no. 12, pp. 737–740, Dezembro 2006.
- [4] Chupeau, B., Oisel, L. e Jouet, P., "Temporal Video Registration for Watermark Detection". *International Conference on Acoustics, Speech and Signal Processing*, v. 2. Toulouse, Maio 2006.
- [5] Carvalho, G., de Oliveira, J. F. L., da Silva, E. A. B., Netto, S. L., et al., "Um Sistema de Monitoramento para Detecção de Objetos em Tempo Real Empregando Câmera em Movimento", *Simpósio Brasileiro de Telecomunicações*, Fortaleza, Setembro 2013.
- [6] Kong, H., Aaudibert, J.-Y., Ponce, J., "Detecting Abandoned Objects with a Moving Camera", *IEEE Transactions on Image Processing*, vol. 19, no. 8, pp.2201–2210, Agosto 2010.
- [7] Diego, F., Ponsa, D., Serrat, J. e López, A., "Video Alignment for Change Detection". *IEEE Transactions on Image Processing*, v.20, no.7, pp.1858–1869, Julho 2011.
- [8] Diego, F., Serrat, J. e López, A., "Joint Spatio-Temporal Alignment of Sequences". *IEEE Transactions on Multimedia*, v.15, no.6, pp.1377–1387, Outubro 2013.
- [9] Klopschitz, M., Zach, C., Irschara, A. e Schmalstieg, D., "Generalized Detection and Merging of Loop Closures for Video Sequences". *International Symposium on 3D Data Processing, Visualization and Transmission*, Atlanta 2008.
- [10] Lovegroove, S. e Davison, A., "Real-Time Spherical Mosaicing Using Whole Image Alignment". *European Conference on Computer Vision*, v.6313, pp. 73–86 Heraklion, Setembro 2010.
- [11] Hall, Brian C., "Lie groups, Lie algebras, and representations: an elementary introduction". 2nd Edition. Springer. 2015.
- [12] MATLAB, version 8.0.0.783 (R2012b), The MathWorks Inc., Natick, Massachusetts, 2012.
- [13] Carvalho, G., Freitas, G., Costa, R., Carvalho, G. et al., "DORIS - Monitoring Robot for Offshore Facilities". *Offshore Technology Conference*, Rio de Janeiro, Outubro 2013.
- [14] Galassi, M., Røyrvåg, A., Carvalho, G. et al., "DORIS - A Mobile Robot for Inspection AND Monitoring OF Offshore Facilities". *Anais do XX Congresso Brasileiro de Automática*, Belo Horizonte, Setembro 2014.
- [15] Freitas, R., Xaud, M., Marcovistz, I., Neves, A. et al., "The embedded electronics and software of DORIS offshore robot". *IFAC Workshop on Automatic Control in Offshore Oil and Gas Production*, Florianópolis, Maio 2015.