Redes Neurais Convolucionais Aplicadas à Preensão Robótica

Renata Oliveira, Errison Alves e Carlos Malqui

DEE - Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), RJ, Brasil
renata.garcia.eng@gmail.com
errison@ele.puc-rio.br
carlos.paucar@hotmail.com

Resumo A preensão robótica é um importante ramo de pesquisa no campo da robótica inteligente que envolve a percepção, planejamento e controle de um manipulador mecânico para efetuar a captura autônoma de objetos. Entretanto, a percepção do ambiente permanece um desafio visto que a forma do objeto pode ser desconhecida, além dos ruídos presentes no ambiente e do elevado custo computacional de processamento. Neste trabalho é proposto o projeto de um modelo computacional, baseado em Redes Neurais Profundas, capaz de detectar uma região de preensão para garras robóticas a partir da imagem RGB do objeto. Na proposta é avaliada o emprego de uma arquitetura mais simples (i.e., menor quantitativo de parâmetros da rede) àquelas comumente utilizadas, buscando-se reduzir o custo computacional do projeto aliado a uma maior capacidade de detecção para objetos inéditos. Dessa forma, foi desenvolvida uma rede composta por menos camadas, propondo também uma redução na dimensão dos parâmetros (resolução de imagem, filtros, etc.). A metodologia proposta foi validada a partir da base de imagens *Cornell Grasp Detection*, a partir da qual o modelo obteve uma acurácia média de 85,3% na detecção da preensão para objetos inéditos, mostrando um desempenho superior ao estado da arte.

Keywords: Preensão Robótica, Imagens RGB, Redes Neurais Profundas

1 Introdução

Os seres humanos, assim como outros animais vertebrados, possuem uma série de reflexos naturais que auxiliam a sobrevivência e o desenvolvimento psicomotor, além dos diferentes sentidos que permitem coletar informação ao interagir com o ambiente. Neste contexto, a visão é o sentido responsável pela percepção e coordenação do corpo durante a execução de alguma atividade [11]. A preensão robótica é um ramo da área de inteligência robótica, voltada à manipulação autônoma de objetos, que busca mimetizar esta habilidade humana natural.

Geralmente a preensão robótica é subdividida em etapas da percepção, planejamento e controle de um manipulador robótico para a captura autônoma de um objeto. O problema é especificado a partir do tipo de manipulador utilizado, da

configuração das garras, bem como das características do objeto a ser capturado [10]. Apesar da mecânica de controle e design dos manipuladores apresentarem grande progresso, a etapa da percepção robótica permanece um grande desafio [16].

Estudos iniciais consideravam o posicionamento das garras robóticas a partir do formato exato do objeto [2]. Porém, esta condição nem sempre pode ser obtida durante situações reais. Dessa forma, outras propostas utilizavam uma simplificação do formato do objeto (esferas, caixas, cones e cilindros), adotando regras heurísticas para a definição das regiões de preensão [12], ou considerando contorno 2D destes objetos [13].

Trabalhos mais recentes [7,17] têm abordado a percepção como um problema de detecção, empregando algoritmos de aprendizado para inferir tais regiões. Além disso, imagens ruidosas ou parciais dos objetos têm sido empregadas para fornecer características mais reais do problema, buscando uma melhor generalização destes modelos de aprendizado. Métodos atuais [10,15] baseados em Redes Neurais Convolutivas propõem o uso de imagens RGB-D dos objetos, utilizando informações de profundidade para melhor inferir regiões de preensão válidas.

Estes modelos convolutivos vêm sendo bastante empregados em sistemas para reconhecimento de imagens, principalmente no campo de visão computacional, possibilitando a proposta de novos sistemas para diferentes tarefas de reconhecimento facial, identificação de dígitos manuscritos, processamento de linguagem natural, entre outros [6].

Neste trabalho é proposto um modelo preditivo, baseado na arquitetura da rede convolutiva Alexnet [9], capaz de detectar uma região de preensão a partir da imagem RGB do objeto. Tal modelo avalia uma redução na dimensão dos parâmetros (camadas, resolução de imagem, filtros etc.), buscando um menor custo computacional de projeto, bem como aumento de detecção da preensão para objetos inéditos.

A estrutura do artigo é a seguinte: inicialmente, a seção 2 apresenta os detalhes do problema. O modelo convolutivo é descrito na seção 3. A seção 4 apresenta a base de dados, a metodologia dos experimentos, os resultados e discussões. Por fim, na seção 5, temos as conclusões e os trabalhos futuros.

2 Descrição do problema

A preensão robótica é inicialmente definida a partir da configuração das garras empregadas no projeto do manipulador. Dentre os tipos mais utilizados se destacam os manipuladores em forma de dedos (2 e 3 dedos), garras paralelas ou *jamming* [10]. Além disso, a percepção do objeto usualmente emprega recursos de imagem 2D, 3D , informações adicionais obtidas a partir de sensores de profundidade, bem como uma combinação de todas estas características descritivas. Nesta proposta são consideradas apenas as informações em 2D do objeto estático (imagem RGB), adotando-se o manipulador de garras paralela.

2.1 Preensão para garras paralelas

Dada a imagem de um objeto, o posicionamento das garras robóticas é representado a partir de cinco informações (Equação 1) que indicam a localização e a orientação da garra paralela antes de capturar o objeto. As coordenadas x, y fornecem a posição central do retângulo de preensão, l representa a abertura das garras robóticas, a indica a faixa de altura onde a garras podem ser posicionadas e θ é a orientação destas em relação ao eixo horizontal, conforme proposto por Lenz et al. [10].

$$p = \{x, y, l, a, \theta\} \tag{1}$$

Na Figura 1, temos uma ilustração dessas informações. Além das coordenadas centrais da região de preensão e da orientação, há a região de atuação das garras em linhas azuis e a distância entre garras, em linhas vermelhas. Ao se utilizar esta representação, o problema da preensão robótica se torna análogo àquele empregado na detecção de objetos pela comunidade de visão computacional. A única diferença consiste no termo adicional referente à orientação das garras.

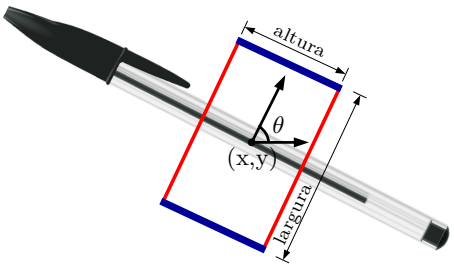


Figura 1: Representação dos cinco parâmetros referentes à localização, ao tamanho e à orientação do retângulo de preensão

3 Modelos Neurais Convolutivas

As Redes Neurais Convolutivas, do inglês *Convolutional Neural Networks* (CNN), são modelos de aprendizado baseados nas diversas áreas do córtex visual e nas relações destas com regiões específicas do campo de visão. Tais regiões, chamadas de campos receptivos, são responsáveis pela ativação de diferentes neurônios, apresentando um nível de sobreposição entre os campos receptivos de neurônios próximos. Este comportamento cerebral motivou a criação de diferentes sistemas para extração de características específicas a partir dos dados.

As CNNs são compostas por diversas camadas que utilizam a operação de convolução para realizar tal extração de características [6]. Esta operação é realizada a partir de uma janela de dados deslizante, chamada de filtro convolutivo, que percorre toda a entrada da rede e opera de forma análoga à sobreposição

dos campos receptivos. O modelo pode conter diversos filtros, cujos valores são ajustados durante o processo de treinamento para a obtenção de características distintas a partir da entrada. Ao final, estas características extraídas se tornam entradas de um algoritmo de aprendizado aplicado à classificação ou regressão, de acordo com o tipo de problema.

É comum entre as camadas convolutivas a realização de outras operações para a redução do espaço de características (*pooling*), normalização e *Zeropadding* [8]. Ao final das camadas convolutivas, geralmente é utilizada uma sequência de camadas de neurônios conectados a todas as ativações das camadas anteriores (*fully-connected*), de forma análoga às camadas das redes neurais tradicionais.

Estes modelos convolutivos têm obtidos bons resultados em diferentes problemas de visão computacional relacionados à classificação e detecção de imagens [5,9], bem como na tarefa da preensão robótica [10,15]. O uso de tais modelos vem sendo impulsionados pelos atuais avanços na capacidade de processamento dos computadores, principalmente com relação ao uso de GPUs.

3.1 Arquitetura proposta

A arquitetura proposta, derivada de uma redução na rede convolutiva Alexnet [9], é composta por 4 camadas convolutivas sequenciais, intercaladas por *pooling*, normalização e *zeropadding* em diversos estágios. Por fim, há 3 camadas *fully-connected* (*Dense*), sendo a última relativa à saída do modelo de regressão logística (*softmax*). A Figura 2 apresenta um descrição completa deste modelo.

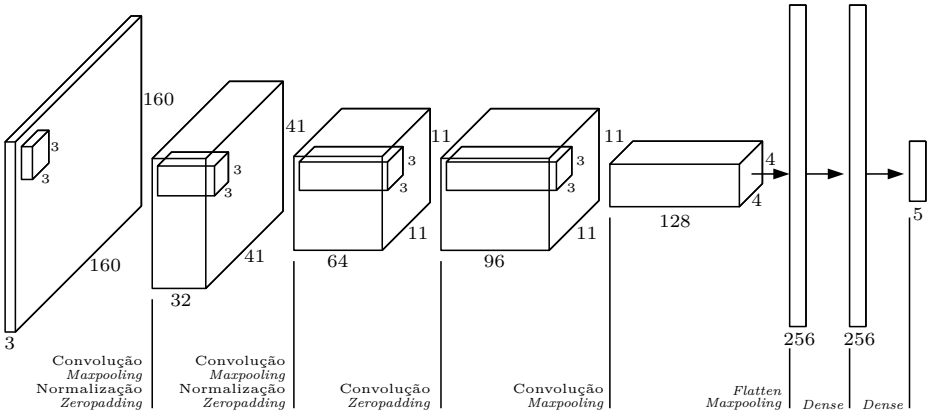


Figura 2: Arquitetura da CNN proposta.

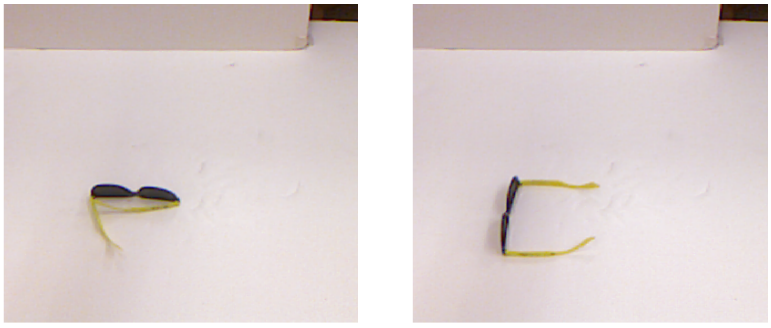
Na Figura 2, estão representadas as dimensões dos filtros convolutivos, adotando um *stride* (passo) de 2 *pixels* tanto para convolução quanto para o *pooling*. Apenas a terceira camada convolutiva considera um passo de 1 *pixel*. Adotou-se o *Maxpooling* para as as camadas *pooling* e a função retificada linear (ReLU)

como função de ativação dos neurônios. Ao final, a camada de saída é composta por 5 neurônios relativos às informações de localização e orientação da garra descritas na seção anterior.

4 Experimentos

4.1 Base de dados

Para o projeto do modelo preditivo, aplicado ao problema da preensão robótica, é necessário que o sistema seja treinado com dados rotulados, ou seja, imagens de objetos e seus respectivos retângulos de preensão. Para tal, utilizou-se a base *Cornell Grasp Detection* [1], que contém 883 imagens RGB rotuladas referentes a 240 objetos diferentes, em que cada objeto pode ser representado em diversas angulações e posições, além de conter múltiplos retângulos de preensão. As Figuras 3a e 3b exemplificam duas imagens originais da base relativas a um objeto.



(a) Imagem 1 do objeto (b) Imagem 2 do objeto

Figura 3: Exemplo de imagens para um objeto da base.

As imagens originais foram recortadas centralmente em 320x320 *pixels*, assegurando a representação total do objeto e suas coordenadas de preensão. Também foi considerada a geração de novos dados a partir de transformações nas imagens, gerando mais entradas para o treinamento do modelo. Para cada imagem da base foram inseridos deslocamentos verticais e horizontais de 50 *pixels* nos diferentes sentidos, resultando na geração de 19.841 diferentes imagens e regiões de preensão associadas. Não foram inseridas rotações, visto que cada objeto já era representado em diferentes angulações. Na Figura 4 temos o resultado do deslocamento inserido na Figura 3b e a representação de uma região de preensão associada.

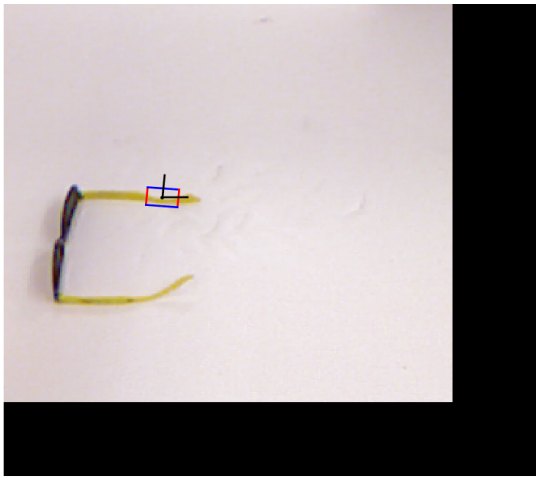


Figura 4: Exemplo das transformações inseridas na base.

4.2 Treinamento

A rede neural convolutiva foi implementada em Python, utilizando a biblioteca Keras [3] aplicada à GPU (nVidia Tesla C2070). Para o treinamento do modelo foi utilizado o algoritmo Nadam padrão [4], um *batch* de 150 imagens, a minimização do erro médio absoluto e um total de 10 épocas. Entre as camadas *fully-connected* foi adotado um *dropout* aleatório de 0.5.

Para a avaliar a predição correta do retângulo de preensão foi adotada a métrica ***rectangle metric*** [7], que compara a orientação e a área entre os retângulos previstos com aqueles definidos na saída do modelo. Inicialmente, o erro de orientação entre as predições e as saídas rotuladas deve ser menor que 30°. Em seguida, avalia-se a sobreposição entre as áreas dos retângulos através do índice de Jaccard [14], considerando suficiente 25% de sobreposição para definir um retângulo válido [15].

O desempenho do modelo foi estimado através da validação cruzada de 5-*folds*, onde o conjunto de treino foi composto pelas imagens geradas a partir dos deslocamentos, enquanto o teste considerou as imagens originais recortadas. Além disso, foram utilizados os critérios *image-wise* e *object-wise* para a divisão das imagens entre os conjuntos treino e teste [7].

No primeiro critério, as imagens são divididas de forma aleatória entre os conjuntos de treino e teste, possibilitando a avaliação do modelo diante de posições diferentes de objetos já conhecidos. A segunda separação considera a divisão dos objetos em conjuntos distintos, ou seja, todas as imagens relativas a um determinado objeto devem estar presente apenas no treino ou teste. Dessa forma, pode-se avaliar a capacidade de predição do modelo diante de objetos inéditos.

Durante o treinamento, diferentes retângulos de preensão foram adotados como saídas para cada imagem, evitando-se que o modelo aprenda a predizer

somente uma forma de preensão para o objeto. Dessa forma, o sistema tende a se adaptar a um valor intermediário dentre todos os retângulos válidos da base.

Na Figura 5, temos um exemplo ilustrativo, em preto e branco, da aplicação dos filtros e operações resultantes de cada camada convolutiva após o treinamento. Pode-se observar as diferentes características extraídas por cada filtro na sequência de camadas sobre a imagem pré-processada da Figura 4. Nota-se também a redução das características provenientes das camadas de *pooling*, bem como o quantitativo de filtros convolutivos empregados em cada etapa.

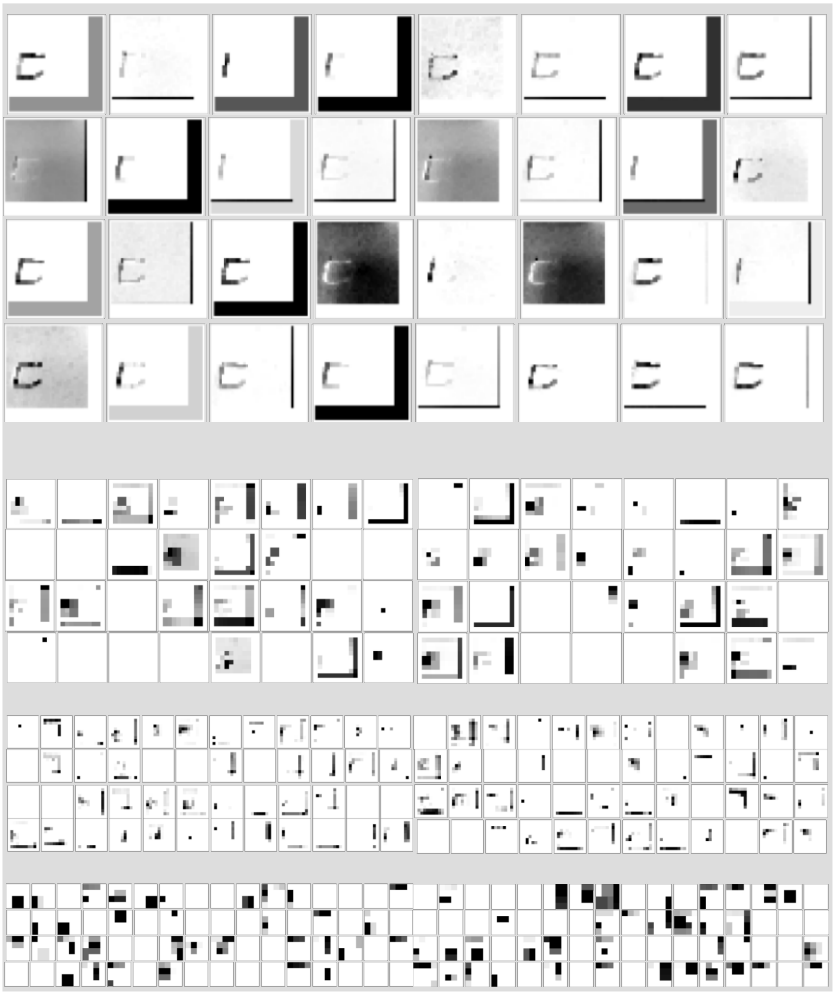


Figura 5: Imagem da Figura 4 após a aplicação dos filtros convolutivos de cada camada.

4.3 Resultados e discussões

Com o modelo convolutivo treinado, a detecção correta da preensão para um objeto é considerada bem sucedida se, ao menos, um dos seus diversos pontos de preensão rotulados for satisfeito pela métrica adotada. A Figura 6 apresenta alguns exemplos de detecção correta. O retângulo verde se refere à predição realizada, enquanto o retângulo preto representa um dos rótulos de preensão do objeto. As linhas que dividem os retângulos indicam a posição central do manipulador robótico.

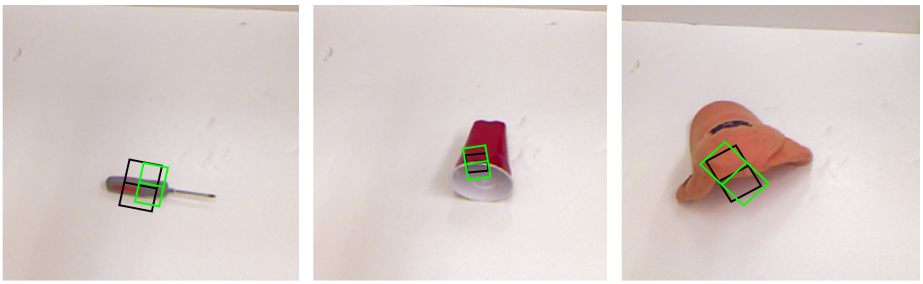


Figura 6: Exemplos de retângulos de preensão previstos corretamente pelo modelo.

A partir da figura, pode-se observar que o modelo consegue predizer retângulos de preensão válidos para objetos de contornos distintos e sob diferentes posições. A região de preensão para objetos com formato longitudinal foi detectada com maior precisão do que em objetos de contornos circular ou de formato irregular. Nos casos de detecção incorreta, a previsão foi frequentemente realizada no meio da seção da imagem que contém o objeto.

Com relação ao desempenho geral do modelo, a Tabela 1 apresenta uma síntese dos resultados obtidos por sistemas de detecção propostos por outros trabalhos utilizando a mesma base de dados. Também é apresentado o desempenho obtido pela arquitetura de 7 camadas (*Convnet7*) proposta na Figura 2. A média e o desvio padrão dos valores apresentados são reportados com relação às épocas de maior desempenho em cada *fold*.

Pela tabela, pode-se observar que a proposta desse artigo é sensivelmente melhor que o modelo proposto por Redmon el al. (2) e superior aos demais para ambos os critérios de separação dos conjuntos de treinamento e teste, considerando a mesma abordagem sob o problema. Dessa forma, nosso modelo demonstra-se apto a ser utilizado tanto para a preensão de objetos em diferentes posições quanto à objetos inéditos.

Cabe observar que as demais propostas comparadas utilizavam informações RGB-D dos objetos, substituindo uma das camadas de cor ou acrescentando uma nova dimensão no dado para representar a profundidade. Além disso, o melhor

Tabela 1: Acurácia dos modelos para o *Cornell Grasping Dataset*

Modelo	*Image-wise*	*Object-wise*
Jiang et al. [7]	60,5%	58,3%
Lenz et al. [10]	73,9%	75,6%
Redmon et al. (1) [15]	84,4%	84,9%
Redmon et al. (2) [15]	85,5%	84,9%
Convnet7	86,5% (±3,0%)	85,3% (±6,3%)

modelo convolutivo de Redmon el al. (2), que realiza a detecção da preensão associada a uma classificação do objeto, foi pré-treinado utilizando outra base de dados.

Em nossa proposta, o modelo convolutivo treinado possui uma arquitetura menor que os demais trabalhos e não foi pré-treinado, reduzindo o tempo de projeto. Sob estas condições, cada época de treinamento utilizando arquitetura de rede empregada por Redmon el al. demorava cerca de 455 segundos para ser finalizada, comparado aos 292s do modelo *Convnet7* proposto. Ou seja, o tempo para o projeto do modelo pôde ser reduzido em até 35%, conforme o *hardware* utilizado.

5 Conclusão

Dado o problema da preensão robótica, neste trabalho foi proposto um sistema para a detecção de regiões para a preensão de objetos a partir do uso de imagens RGB . O modelo *Convnet7* proposto foi baseado em CNNs, avaliando-se uma redução da arquitetura do Alexnet, de forma a projetar um modelo eficaz e de menor custo computacional para o problema avaliado. A proposta se mostrou superior aos demais métodos do estado da arte e se mostrou capaz predizer a localização e orientação de manipuladores de garras paralelas com uma média de 85,3% de acerto, considerando o caso mais complexo (*object-wise*) para a base *Cornell Grasping Dataset*.

Os diferentes experimentos realizados durante o projeto demonstraram que o ajuste fino do modelo foi fundamental para a obtenção de melhores desempenhos, além de simplificar a arquitetura, considerando o escopo de aplicação e complexidade do problema. Para esta aplicação, pode-se concluir que não há a necessidade de um detalhamento excessivo dos objetos, visto que as características dos contornos se mostram mais relevantes do que os detalhes internos do objeto.

Por fim, pretende-se avaliar o desempenho deste modelo imagens de objetos de outros bancos de dados, propor novos ajustes na arquitetura da rede e estender sua aplicação para problemas diversos. Além disso, dada a sensibilidade observada no desempenho do modelo diante de diferentes características nos contornos dos objetos, será realizada uma análise mais detalhada deste efeito e a

relação com sua a eficiência geral. Outra proposta futura seria avaliar a redução no quantitativo de camadas de cores da imagem, considerando uso de imagens monocromáticas neste problema de preensão robótica.

Agradecimentos. As orientações dos professor Leonardo Mendonza e ao colega Cristian Muñoz pelo suporte técnico com as simulações nos computadores do ICA/DEE/PUC-Rio. A CAPES e PUC-Rio, pelo apoio financeiro e de infraestrutura concedido.

Referências

1. Cornell grasping dataset, `http://pr.cs.cornell.edu/grasping/rect_data/data.php`
2. Bicchi, A., Kumar, V.: Robotic grasping and contact: A review. In: Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on. vol. 1, pp. 348–353. IEEE (2000)
3. Chollet, F., et al.: Keras (2015), `https://github.com/fchollet/keras`
4. Dozat, T.: Incorporating nesterov momentum into adam (2016)
5. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 580–587 (2014)
6. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016), `http://www.deeplearningbook.org`
7. Jiang, Y., Moseson, S., Saxena, A.: Efficient grasping from rgbd images: Learning using a new rectangle representation. In: Robotics and Automation (ICRA), 2011 IEEE International Conference on. pp. 3304–3311. IEEE (2011)
8. Karpathy, A.: Stanford university cs231n: Convolutional neural networks for visual recognition, `http://cs231n.stanford.edu/`
9. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
10. Lenz, I., Lee, H., Saxena, A.: Deep learning for detecting robotic grasps. The International Journal of Robotics Research 34(4-5), 705–724 (2015)
11. de Mattos Ferreira, C.: Psicomotricidade: da educação infantil à gerontologia - teoria e prática. Lovise (2000)
12. Miller, A.T., Knoop, S., Christensen, H.I., Allen, P.K.: Automatic grasp planning using shape primitives. In: Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on. vol. 2, pp. 1824–1829. IEEE (2003)
13. Morales, A., Sanz, P.J., Del Pobil, A.P.: Vision-based computation of three-finger grasps on unknown planar objects. In: Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on. vol. 2, pp. 1711–1716. IEEE (2002)
14. Real, R., Vargas, J.M.: The Probabilistic Basis of Jaccard's Index of Similarity. Systematic Biology 45(3), 380–385 (1996)
15. Redmon, J., Angelova, A.: Real-time grasp detection using convolutional neural networks. In: Robotics and Automation (ICRA), 2015 IEEE International Conference on. pp. 1316–1322. IEEE (2015)
16. Yoshikawa, T.: Multifingered robot hands: Control for grasping and manipulation. Annual Reviews in Control 34(2), 199–208 (2010)

17. Zhang, L.E., Ciocarlie, M., Hsiao, K.: Grasp evaluation with graspable feature matching. In: RSS Workshop on Mobile Manipulation: Learning to Manipulate (2011)