VideoRecognition - Uma proposta de serviço para reconhecimento de elementos de vídeo em larga escala

Antonio J. G. Busson
TeleMídia/PUC-Rio
busson@telemidia.puc-rio.br

Álan L. V. Guedes
TeleMídia/PUC-Rio
alan@telemidia.puc-rio.br

Gabriel N. P. dos Santos
Faculdade ISL | Wyden
gabrieainha@gmail.com

Carlos de Salles Soares Neto
TeleMídia/UFMA
csalles@deinf.ufma.br

Ruy Luiz Milidiú
LEARN/PUC-Rio
ruy@inf.puc-rio.br

Sergio Colcher
TeleMídia/PUC-Rio
colcher@inf.puc-rio.br

ABSTRACT

Deep Learning research has allowed significant advancement of various segments of multimedia, especially in tasks related to speech processing, hearing and computational vision. However, some video services are still focused only on the traditional use of media (capture, storage, transmission and presentation). In this paper, we discuss our ongoing research towards a DLaS, *i.e.* Deep Learning as a Service. This way, we present the state of art in video classification and recognition. Then we propose the *VideoRecognition* as DLaS to support the tasks such as: image classification and video scenes, object detection and facial recognition. We discuss the usage of the proposed service in the context of the video@RNP repository. Our main contributions consist on dicussussions over the state of art and it usage in nowdays multimedia services.

KEYWORDS

Video Recognition, Deep Learning, CNN, Video reposistories

1 INTRODUÇÃO

Nos últimos anos, o *Deep Learning* (DL) permitiu significativo avanço de vários segmentos da multimídia, principalmente em tarefas relacionadas a processamento de fala, audição e visão computacional [17]. Plataformas como IBM Watson[1] e Microsoft Azure ML[2] já oferecem DLaS (*Deep Learning as a Service*), permitindo que sistemas multimídia (e.g. Helpicto [5], PersonalizedTV [15], ShrewsburyMuseum [11]) possam incorporar novas funcionalidades baseadas em aprendizagem e reconhecimento de padrões, o que os leva para a categoria de IIMS (*Intelligent Interactive Multimedia Systems*).

Na contramão, muitos serviços multimídia usados no Brasil (e.g. Video@RNP[3], Videoaula@RNP[4]) ainda estão focados apenas no uso tradicional das mídias (captura, armazenamento, transmissão e apresentação). O uso de um serviço de reconhecimento está fortemente ligado a melhora na busca e indexação do conteúdo através da descoberta de metadados mais precisos, mas também pode ser útil para a criação de funcionalidades sofisticadas, como por exemplo: enriquecimento de conteúdo, reconhecimento facial, monitoração de *streaming*, detecção de conteúdo impróprio, etc. Nesse sentido, é interessante que exista um serviço dedicado à aprendizagem e reconhecimento de conteúdo multimídia, que tenha baixo custo e que mantenha o sigilo dos dados.

Neste artigo, propomos o *VideoRecognition*, um serviço para reconhecimento de elementos de vídeo em larga escala. O *VideoRecognition* faz uso de tecnologias abertas de *Deep Learning* para suportar tarefas de: classificação de imagem e cenas de vídeo, detecção de objetos e reconhecimento facial. A arquitetura do serviço foi planejada para cumprir três requisitos básicos: (1) permitir que analistas possam especificar arquiteturas de DL e padrões de dados para tarefas de reconhecimento específicas; (2) oferecer uma ferramenta para que usuários possam anotar conteúdo multimídia e gerar *datasets*; (3) oferecer uma interface para integração com outros serviços multimídia. Dessa forma, possibilitando que outros serviços possam incorporar funcionalidades baseadas em reconhecimento automático.

O restante desse artigo está organizado como segue. A Seção 2 descreve o *estado-da-arte* nas tarefas de classificação de imagem e vídeo, detecção de objetos e reconhecimento facial. Em seguida, a Seção 3 apresenta a proposta deste artigo. A Seção 4 apresenta um cenário de uso para integração da proposta com o serviço Video@RNP. Por fim, a Seção 5 contém as considerações finais.

2 ESTADO DA ARTE

Métodos baseados em *Deep Learning* se tornaram o estado da arte em vários segmentos da área de sistemas multimídia. Nesta seção, são apresentados os modelos baseados em *Deep Learning* que já são considerados maduros suficientes para o desenvolvimento de serviços de reconhecimento de elementos de vídeo. Na Subseção 2.1 é apresentado o *estado-da-arte* para classificação de imagem e vídeo. Em seguida, na Subseção 2.2 é apresentado o *estado-da-arte* para as tarefas de detecção de objetos e reconhecimento facial.

2.1 Classificação de imagem e vídeo

Na multimídia a tarefa de classificação consiste em mapear um conteúdo de mídia em uma ou mais categorias distintas. Arquiteturas de *Deep Leaning* baseadas em CNNs (*Convolutional neural network*) ou ConvNets se tornaram o principal método usado para reconhecimento de padrões áudio-visuais. Tipicamente o treinamento de

[1] https://www.ibm.com/watson/
[2] https://azure.microsoft.com/pt-br/services/machine-learning-studio/
[3] http://video.rnp.br/portal/home
[4] http://www.videoaula.rnp.br/portal/home

In: V Workshop "O Futuro da Videocolaboração" (WCT-Video 2018), Salvador, Brasil. Anais do XXIV Simpósio Brasileiro de Sistemas Multimídia e Web: Workshops e Pôsteres. Porto Alegre: Sociedade Brasileira de Computação, 2018.

ISBN 978-85-7669-435-9.

CNNs é feito de maneira supervisionada, e são treinadas em *datasets* que contém milhares/milhões de mídias e classes relacionadas. Durante o treinamento, as CNNs aprendem a hierarquia de *features* que são aplicadas a mídia de entrada para que seja possível realizar a classificação do seu conteúdo.

Desde a vitória da CNN AlexNet [12] no desafio ImageNet 2012 [20], surgiram novas arquiteturas baseadas em CNN cada vez mais precisas. A vencedora do ImageNet 2014, por exemplo, foi a CNN InceptionNet [23], que propôs o uso do bloco Inception, um bloco que usa vários filtros de diferentes tamanhos no mesmo nível para resolver o problema de localização da informação em imagens. No ano seguinte, a rede ResNet [7] foi a vencedora do ImageNet 2015, e introduziu o conceito de conexões residuais, que aumentou a performance e reduziu o tempo de treinamento das CNNs. Mais tarde foi desenvolvida a arquitetura Inception-Resnet [22], que combina os blocos Inception com as conexões residuais. Essa arquitetura é popular e é a base para muitos outras arquiteturas de CNN para extração de *features*.

A arquitetura SE-Net (*Squeeze-and-Excitation Network*) [10] é o *estado-da-arte* na tarefa de classificação de imagens, obtendo 2.25% de erro top-5 no ImageNet 2017. Assim como o InceptionNet, a SE-Net propões um novo tipo de bloco chamado SE, que melhora o poder de representação da rede ao destacar as interdependências entre os canais da imagem e seus mapas de *features*. Para isso, a SE-Net usa um mecanismo que permite que a rede possa fazer uma recalibração de *features*, através do qual, usa informações globais para enfatizar as *features* mais informativas e suprimir as *features* menos úteis.

Diferente de imagens, vídeos não possuem apenas informação visual, mas também auditiva. Métodos atuais para classificação de vídeo geralmente representam os vídeos pelas *features* audiovisuais extraídas dos seus quadros, seguido pela agregação dessas *features* sobre o tempo. Métodos usados para extração de features incluem modelos de CNN pré-treinados em *datasets* de larga escala. O YouTube8M [1], por exemplo, usa a rede Inception-Resnet pré-treinada no ImageNet para extrair *features* visuais, e usa a rede Audio-VGG [8] pré-treinado no Audioset [6] para extrair *features* auditivas. Já métodos para agregação de *features* incluem métodos sofisticados de *pooling* como na CNN NetVLAD [2] ou usando modelos recorrentes baseados em GRU [3] ou LSTM [9]. A arquitetura Gated-NetVLAD [16] conseguiu obter 84% de GAP no desafio Youtube8M 2017[5], que contém mais de 6 milhões de vídeos distribuídos em 3862 classes de etiquetas.

2.2 Detecção de objetos e reconhecimento facial

A tarefa de detecção de objetos consiste em localizar e classificar objetos que estão dentro da mídia. A arquitetura YOLO (*You Only Look Once*) [18] é considerada o *estado-da-arte* na tarefa de detecção de objetos. Sua última versão, chamada YOLOv3 [19] obeteve um mAP de 57.9% no dataset COCO (*Common Objects in Context*) [14]. O YOLO é ideal para aplicações de vídeo em tempo-real, visto que é o modelo de detecção de objetos baseado em CNN mais rápido da literatura, chegando a rodar próximo de 30 FPS na GPU Pascal Titan X[6]. O YOLO divide a imagem em várias regiões e prediz uma caixa delimitadora de objeto para cada região. As coordenadas da caixa delimitadora podem ser usadas para destacar ou segmentar o objeto da imagem. As imagens segmentadas, por sua vez, podem ser usadas em outros tipos de tarefa, como por exemplo, de reconhecimento facial.

A tarefa de reconhecimento facial tenta responder a pergunta "Quem é esta pessoa?". Tipicamente, a identificação de imagens de face é feita pela comparação da similaridade entre uma ou mais imagens de indivíduos já conhecidos. A arquitetura FaceNet [21] é o *estado-da-arte* para a tarefa de reconhecimento facial. Esta arquitetura tem uma acurácia de 99.6% no *dataset* LFW (*Labeled Faces in the Wild*) [13]. Dada duas imagens de face, o FaceNet extrai as *features* faciais a partir da ativação linear da última camada densa da rede Inception-Resnet. Em seguida, a similaridade entre as imagens de face é calculada pela distância euclidiana entre os vetores de *features*, se a distância for menor que um limiar, então presume-se que as imagens de face sejam da mesma pessoa. Ao usar o FaceNet em conjunto com o YOLO é possível realizar reconhecimento facial em vídeo e *live streaming*.

3 PROPOSTA DE TRABALHO

Nesta seção é apresentado o *VideoRecognition*, um serviço que visa oferecer interfaces para anotação de conteúdo e reconhecimento de elementos audiovisuais. A ideia é permitir que outros serviços multimídia possam utilizar essas interfaces para incorporar funcionalidades mais sofisticadas a suas próprias arquiteturas. Inicialmente o *VideoRecognition* é restrito a três tipos de tarefas: classificação de imagem e cenas de vídeo, detecção de objetos e reconhecimento facial. Adiante é apresentada a arquitetura do *VideoRecognition*, e em seguida, uma discussão de como outros serviços multimídia podem se beneficiar das funcionalidades do *VideoRecognition*.

A arquitetura do *VideoRecognition* é ilustrada na Figura 1. O grupo *Client Services* contém serviços que utilizam o *VideoRecognition*, como exemplos citamos: Um repositório de Vídeo (*RV*), um serviço de conferência de vídeo (*VideoConference*), e por último, um editor de vídeo (*VideoEditor*). O *VideoRecognition* possui duas interfaces de acesso: *annotationIF* e *recognitionIF*. A interface *annotationIF* permite que os serviços possam alimentar o *VideoRecognition* com mídias anotadas, em outras palavras, permite a criação de *datasets*. Posteriormente os *datasets* são utilizados para o treinamento dos modelos de reconhecimento. Por sua vez, a interface *recognitionIF* permite que os serviços possam utilizar os modelos de reconhecimento já treinados.

O *VideoRecognition* contém uma pilha de modelos, onde cada um é especializado em um tipo de tarefa e dedicado a um serviço externo. Esses modelos são projetados por analistas que especificam a arquitetura de DL (dentre as descritas na Seção 2), as classes, tipo de tarefa e qual serviço pode utilizá-lo. Ao acessar uma interface, o serviço cliente deve especificar qual modelo da pilha deve ser utilizado. Cada modelo da pilha possui dois sub-módulos: *train* e *recognize*. O módulo *train* usa as informações do *dataset* para realizar o treinamento da arquitetura, por sua vez o módulo *recognize* utiliza os pesos do treinamento para realizar as predições solicitadas através da interface *recognitionIF*.

[5] https://research.google.com/youtube8m/

[6] https://www.nvidia.com/pt-br/geforce/products/10series/titan-x-pascal/

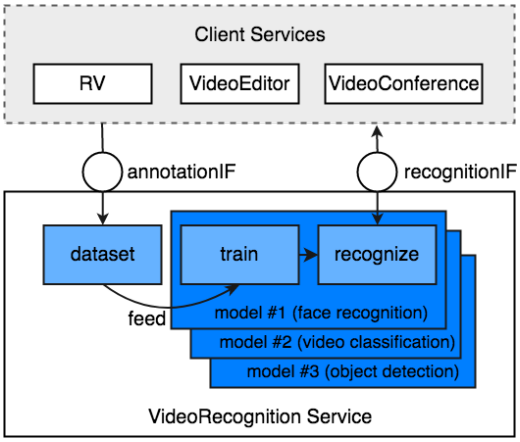

Figura 1: Arquitetura do *VideoRecogtion*.

Ao utilizarem o *VideoRecognition*, os *Client Services* podem prover, dentre outras, as seguintes funcionalidades (F):

- **F.1. Busca utilizando mídia de referência.** Essa funcionalidade visa permitir *usuários de um Client Service proativamente encontrar conteúdos de mídias que pertencem a uma mesma classificação*. Nela, o usuário utiliza uma conteúdo de mídia como parâmetro de busca. Em um RV, por exemplo, o usuário pode inserir a foto da Grécia antiga e, em seguida, o *VideoRecognition* pode listar vídeo aulas sobre a Grécia antiga. Adicionalmente, um criador de vídeo utilizando um *VideoEditor* pode visualizar quais outros conteúdos presentes em um RV são relacionados ao vídeo que está editando. Isso permite enriquecer o seu conteúdo de vídeo, seja com novos links, ou novos conteúdos de mídias.
- **F.2. Reconhecimento facial.** Essa funcionalidade visa permitir *usuários de um Client Service proativamente encontrarem conteúdos de mídias que possuem a mesma pessoa*. Nela, o usuário utiliza um conteúdo de mídia com pessoa específica como parâmetro de busca. Em um RV, por exemplo, o usuário pode inserir a foto de um professor específico para o RV listar vídeos onde aparece o professor desejado.
- **F.3. Recomendação.** Essa funcionalidade está relacionada com a capacidade de um *Client Service automaticamente listar conteúdos de mesma classificação ou com uma mesma pessoa* para seus usuários. Em uma RV, por exemplo, a visualização de um vídeo pode ser acompanha com a lista de vídeos relacionados.
- **F.4. Detectar conteúdo impróprio.** Essa funcionalidade visa prevenir a presença de conteúdos impróprios em *Client Services*. Essa funcionalidade requisita um treinamento prévio de classificação de conteúdos que são considerados impróprios (*e.g.*, cenas de sexo, uso de drogas, gore). Um *Client Service* pode solicitar que o *VideoRecognition* analise o conteúdo da mídia para verificar se o conteúdo é impróprio.
- **F.5. Busca por trecho de interesse.** Essa funcionalidade é relacionada com a busca ou segmentação temporal de trechos de vídeo segundo um critério de filtragem estabelecido pelo usuário de um *Client Service*. Em um *VideoEditor*, por exemplo, o usuário pode solicitar que o *VideoRecognition* segmente todos os trechos do vídeo onde aparecem paisagens. Em um cenário de RV, o usuário pode fazer buscas por trechos de vídeo específicos, onde o *VideoRecognition* pode, por exemplo, listar apenas cenas de vídeo onde aparecem carros esportivos.
- **F.6. Sincronismo com eventos de reconhecimento.** Essa funcionalidade está relacionada com a capacidade de sincronizar elementos de mídia com eventos de reconhecimento. Nesse caso, o usuário de um *Client Service* pode estabelecer regras de reconhecimento para iniciar ou terminar a apresentação de uma ou mais mídias. Por exemplo, durante a transmissão de uma palestra pelo *VideoConference*, uma janela com fotos e texto biográfico sobre o palestrante podem estar disponíveis enquanto o palestrante estiver em cena. o *VideoRecognition* pode notificar o serviço *VideoConference* sobre presença e identidade do palestrante para que o conteúdo multimídia correto seja apresentado.

4 CASO DE USO NO VIDEO@RNP

Com o propósito de avaliar a proposta, nesta seção é apresentado um cenário de uso para integração do *VideoRecognition* com um serviço existente. Mais precisamente, propomos a integração com o RV vídeo@RNP. Segundo Ciuffo *et. al.* [4], o vídeo@RNP é um serviço para disponibilização e armazenamento de conteúdo audiovisual de instiuções acadêmicas que tem como diferencial sua infraestrutura de rede, um tipo de CDN (Content Delivery Network) restrita ao tráfego audiovisual. Em particular, inicialmente é proposto que esse serviço seja estendido com as funcionalidades **F.2** e **F.3**.

Relacionado a **F.2**, identifica-se que uma das limitações desse serviço é indexação de vídeos, e consequentemente a sua busca, de conteúdo. Por exemplo, o vídeo[7] do professor Marcos Bagno sobre gramática realizado na Faculdade de Ciências e Letras (FCL) da UNESP possui as *keywords*: "Marcos", "Bagano", "FCL"e "gramática". Percebamos que o professor não está corretamente indexado. Acredito que essas *keywords* tenham sido retiradas automaticamente da descrição do vídeo. Essa limitação poderia ser contornada com o uso do *VideoRecognition*.

O video@RNP poderia acessar o módulo de `recognition` para realizar a indexação de pessoas nos vídeos. Entretanto, o `recognition` precisa ser treinado para identificar pessoas específicas (*e.g.* quem é Marcos Bagno). Para isso, primeiro o vídeo@RNP necessita solicitar ao módulo `recognition` para segmentar faces. Em seguida, através de uma interface de anotação, um usuário de administração do vídeo@RNP poderia anotar faces reconhecidas e enviar para o módulo `annotation`. A Figura 2 ilustra um *mockup* dessa interface. Ela indica o vídeo analisado e a segmentação de pessoas durante a apresentação do vídeo (Figura 2-A). Adcionalmente, ela apresenta uma lista de reconhecimentos de faces (Figura 2-B). Tais reconhecidos podem indicar novas faces não conhecidas (Figura 2-C) ou faces já conhecidas que necessitam de confirmação (Figura 2-D).

Relacionado a **F.3.**, identifica-se que o vídeo@RNP realiza apenas uma busca facetada, ou seja, usuários apenas navegam em conjuntos de vídeos definidos pelas *keywords* inseridas na criação dos vídeos.

[7] http://video.rnp.br/portal/video.action?idItem=26242

Esse serviço poderia se utilizar do modulo `recongnition` para gerar uma lista de vídeos relacionados a semântica do conteúdo ao vídeo visualizado. O critério semântica poderia pode a classificação do vídeo (*e.g.* aulas de gramáticas) ou a presença de uma pessoa específica (*e.g.* professor Marcos Bagno).

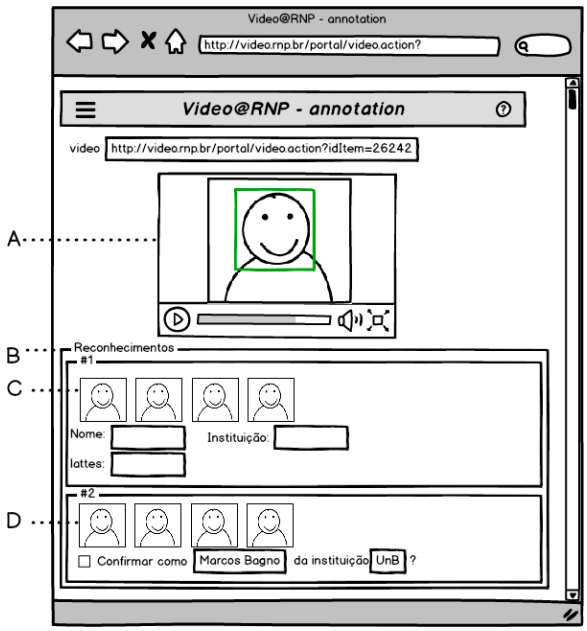

Figura 2: Esboço da interce de anotação do video@RNP com suporte do *VideoRecognition*

5 CONSIDERAÇÕES FINAIS

Este trabalho apresenta uma visão geral do estado da arte de atividades de reconhecimento de elementos audiovisuais em vídeos. Com o objetivo de preencher a lacuna desses atividades em serviços certos serviços multimídia, apresentamos o *VideoRecognition* como sendo uma proposta de DLaS (Deep Learning as a Service). Ele visa oferecer suporte a outros serviços de vídeo como RV, editor de vídeo e vídeo conferência. Nesse sentido, apresentamos uma visão arquitetural do *VideoRecognition* e discutimos para estender outros serviços de vídeo. Em particular, discutimos um caso de uso sobre RV chamado video@RNP e apresentamos um *mockup* da ferramenta de anotação. Nossas principais contribuições consistem em discussões sobre o estado da arte e no uso do *VideoRecognition* em serviços multimídia atuais.

O serviço faz parte de uma pesquisa em andamento. Citamos como trabalhos futuros: (a) explorar novos métodos de *Deep Learning* dedicados a outras modalidades de mídia que ainda não são suportadas pelo *VideoRecognition*, como áudio e legendas de vídeo; (b) enriquecer o *VideoRecognition* com novos tipos de tarefas, como por exemplo, detecção de atividades em vídeo; (c) promover sessões de design participativo com desenvolvedores e administradores de outros serviços multimídia a fim de identificar e validar requisitos para o *VideoRecognition*.

REFERÊNCIAS

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Apostol (Paul) Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. 2016. YouTube-8M: A Large-Scale Video Classification Benchmark. In *arXiv:1609.08675*. https://arxiv.org/pdf/1609.08675v1.pdf

[2] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. 2016. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5297–5307.

[3] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259* (2014).

[4] Leandro Ciuffo, Marcelino Cunha, Christian Miziara de Andrade, Graciela Leopoldo Martins, Jean Carlo Faustino, Rafael Valle, Fausto Vetter, Antônio Carlos Nunes, and Helder Vitorino. Mapeamento de soluções de videocolaboração da RNP. In *Anais do XXIII Simpósio Brasileiro de Sistemas Multimídia e Web: Workshop do CT-Vídeo (Comitê Técnico de Prospecção Tecnológica em Videocolaboração)*,. 337–368.

[5] Equadex. 2018. Helpicto. (2018). http://www.helpicto.com/ Accessed: 2018-05-18.

[6] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio set: An ontology and human-labeled dataset for audio events. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 776–780.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[8] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. 2017. CNN Architectures for Large-Scale Audio Classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://arxiv.org/abs/1609.09430

[9] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[10] Jie Hu, Li Shen, and Gang Sun. 2017. Squeeze-and-excitation networks. *arXiv preprint arXiv:1709.01507* 7 (2017).

[11] Martin Kearn and Martin Beeby. 2017. Using Cognitive Services to make museum exhibits more compelling and track user behavior. (2017). https://microsoft.github.io/techcasestudies/cognitive%20services/2017/0\8/04/BlackRadley.html Accessed: 2018-05-18.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[13] Gary B. Huang Erik Learned-Miller. 2014. *Labeled Faces in the Wild: Updates and New Reporting Procedures*. Technical Report UM-CS-2014-003. University of Massachusetts, Amherst.

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

[15] Julio Wong Mathews Thomas, Janki Vora and Satish Sadagopan. 2016. Use cases for industry cognitive solutions. (2016). https://www.ibm.com/developerworks/library/cc-cognitive-media-telco-2-trs/index.html Accessed: 2018-05-18.

[16] Antoine Miech, Ivan Laptev, and Josef Sivic. 2017. Learnable pooling with Context Gating for video classification. *arXiv preprint arXiv:1706.06905* (2017).

[17] Kaoru Ota, Minh Son Dao, Vasileios Mezaris, and Francesco GB De Natale. 2017. Deep learning for mobile multimedia: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 13, 3s (2017), 34.

[18] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 779–788.

[19] Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).

[20] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115, 3 (2015), 211–252.

[21] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 815–823.

[22] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning.. In *AAAI*, Vol. 4. 12.

[23] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1–9.