CNN-based Fall Detection using Optical Flows

ML Singapore! Group 4
Li Keyou (A0206174L), Maxx Chan (A0206170U), Sun Hao Ting (A0206156L),
Ek Chin Hui (A0205578X), Tay Yu Hong (A0189568A), Xu Ziyi (A0204675A)

1. Introduction

With one of the most rapidly ageing populations in the world (Bloomfield 2019), Singapore is estimated to have about 32% of its population aged 65 and above by 2035. It is estimated that 83,000 elderly persons will be living alone by 2030 (Ng 2020a). With the younger population working an average of 44.7 hours a week (Hirschmann 2020), along with our high and increasing old-age dependency ratio of 21.6 (Hirschmann 2021), the **elderly are vulnerable and lack support.**

According to the Ministry of Health, about **one-third of elderly aged 60 and above have fallen more than once** (HealthHub 2021). With their impaired mobility, it can be difficult for them to get up on their own. If immediate medical care is not given, they could potentially suffer complications such as "pressure sores, carpet burns, dehydration, hypothermia, pneumonia, and even death" (Fleming and Brayne 2008). Hence, **timely response is of upmost importance.**

Given the shortage of professional caregivers like nurses (Hermesauto 2016) in Singapore, a **more automated system may prove useful**. To that end, our team aims to develop a fall detection model which takes in video stream data as input, to alert the relevant support networks and help the elderly receive timely aid. We envision this being deployed in hospitals, nursing homes, and perhaps homes of at-risk elderly.

2. Existing Methods

The existing approaches to detect falls can be grouped into the following: wearable sensor-based (Karantonis et al. 2006), ambient sensor-based (Zhang and Karunanithi 2016) and computer vision-based methods (Rougier et al. 2006).

2.1 Wearable Sensor-based Fall Detection Devices

Fall detection wearable devices such as wristbands or necklaces (Senior Care Singapore 2020) determine falls based on accelerometer data. The greatest barriers to their widespread use are inconvenience and privacy (GovTech 2020). Some elderly people may be unwilling or unable to remember to wear these devices The recent uproar over the TraceTogether token (for COVID-19 contact tracing) is also testament to privacy and data usage concerns regarding wearable tracking devices (Chia 2021). In addition, public datasets of fall data collected from wearable devices require intrusive chest motion sensors, which is even less likely to be adopted by elderly.

2.2 Ambient Sensor-based Fall Detection

Vibration or pressure sensors are the most common methods adopted by ambient sensor-based detection. These sensors are usually installed on the floor surface or under the bed (Zigel, Litvak, and Gannot 2009) in the homes of elderly. As compared to wearable devices, these devices are less costly and less of a hassle for users (Mirmahboub et al. 2013), once set up. However, the detection rate is significantly lower and suffer from high false positive rates (Wang et al. 2015).

2.3 Vision-Based Fall Detection

Given the challenges with other approaches, vision-based fall detection is a promising area of research, and may potentially be a low-cost solution by tapping upon existing infrastructures such as closed-circuit television footage (CCTV) cameras. The current approaches of vision-based fall detection usually involve the usage of depth cameras to recognise human motion. To avoid the cost of a specialised depth camera, we seek to explore vision-based fall detection using normal input videos.

2.4 Vision-Based Fall Detection with Optical Flows and Convolutional Neural Networks (Núñez et al. (2017))

An interesting vision-based model is from Núñez et al. (2017). In this model, the input video is pre-processed into optical flow images, afterwhich the model is trained via a three-step training phase, consisting of a VGG-16 feature extractor and a fully connected neural network (FC-NN) classifier for classification. (See Figure 1)

In their case, Núñez et al. (2017) trained the neural network on the UR Falls Dataset (URFD) (Kwolek and Kepski 2014), Multiple Cameras Fall Dataset (Multicam) (Rougier et al. 2006), and Fall Detection Dataset (FDD) (Adhikari, Bouchachia, and Nait-Charif 2017).

To better detect falls, the optical flow algorithm was used to represent motion effectively and minimise the influence of environmental features. Optical flow is used to describe the displacement vector fields between two consecutive frames,

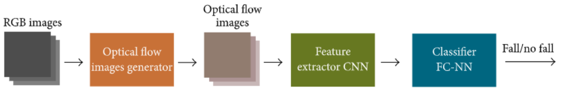


Figure 1: System architecture, or pipeline of Núñez et al.'s (2017) implementation.

and static features of the images are removed. Frames of optical flow images fed to the model in stacks of 10 as input into the CNN (See Figure 2), to effectively capture the duration of a fall.

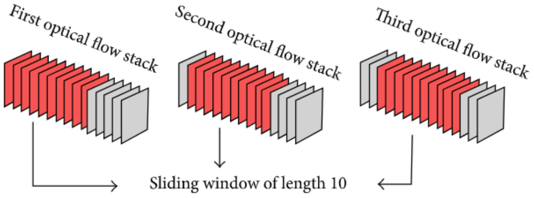


Figure 2: Sliding window implementation

The horizontal and vertical displacement vector fields were combined in 1 stack to create stacks of $R^{224\times224\times2L}$, where L = 10, the number of frames in 1 stack.

The original VGG-16 network was trained using the Imagenet dataset which has over 14 million images and 1000 classes, followed by the UCF101 dataset which contains 13,320 videos including 101 human actions. Input was also changed to stacks of optical flow images $R^{224\times224\times2L}$. Finally, the convolutional layers' weights were frozen and arrays of extracted features of $F \in R$ of size 4,096 for each input stack were fed into a FC-NN classifier, in order to fine tune the network for fall detection. Finally, the classifier will output the classification of "fall" or "no fall".

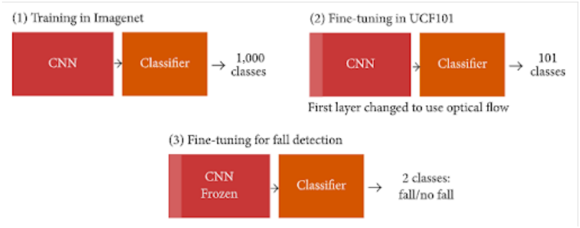


Figure 3: Transfer Learning process of the neural network, adapted from Núñez et al. (2017)

The binary cross-entropy loss function defined as follows was chosen:

$$loss(p,t) = -(t \cdot log(p) + (1-t) \cdot log(1-p)) \quad (1)$$

[p: prediction of the network, t: ground truth]

As the network did not perform well for the "not fall" class, it was modified by adding a class weight to the loss function to increase the importance of the fall class as shown below:

$$loss(p,t) = -(w_1 \cdot t \cdot log(p) + w_0 \cdot (1-t) \cdot log(1-p)) \quad (2)$$

A w_0 greater than 1.0 is used to penalize the loss function for every mistake made on the "fall class" more than the "no fall class". Backward propagation algorithm is adopted here to minimise the modified binary cross-entropy loss function, thus the model is biased towards falls. This is reasonable as an undetected fall has more severe consequences than a false positive.

3. Goals of Product

We endeavour to build upon the work of Núñez et al. (2017) to create a fall detection system to be deployed in senior care facilities in Singapore such as assisted living estates, hospitals and nursing homes.

Given the shortage of care personnel in primary care facilities, our vision-based fall detection system could be used to support care personnel in responding faster to elderly falls. This can be achieved by running video stream input from cameras in common areas or rooms with lower human traffic through our system, then alerting caregivers via a telegram bot message when a fall happens.

One of the goals in Singapore's Action Plan for Successful Ageing is to "Age in Place" (Zainal and Teoh 2018) - to retain autonomy and independence over our lives even as we grow older and face the challenges of old age. One notable development is Singapore's upcoming first assisted living HDB estate in Bukit Batok (Ng 2020b). The concept is centred around letting seniors live independently, whilst having access to services such as housekeeping, home fixes, and 24-hour emergency monitoring and response. Our system could be a solution to alert caregivers to elderly residents who suffer from falls, thus achieving timely response without having to assign personnel to supervise seniors. In essence, our system can help the facility manage risk better as they strive to provide seniors with a more independent living environment.

4. Methodology

Our project builds upon the work of Núñez et al. (2017) - training a CNN with the UR Falls Dataset. The full pipeline of the system can be visualised as below in Figure 4.

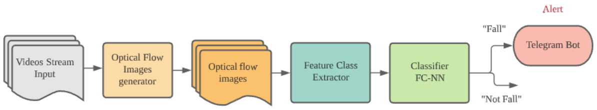


Figure 4: The system architecture or pipeline: the video stream input is converted to optical flow images, then feature classes are extracted with a CNN, and a FC-NN classifies whether there has been a fall or not, which a signal will be sent to the telegram bot in the case of a fall, and the bot will send an alert.

4.1 Data-Processing: Reduce Sampling Rate of Sliding Window

One potential source of inaccuracy we noticed with Núñez et al.'s (2017) data processing method was that all stacks generated from 1 video, x, would have the same truth label c(x). This meant that a set of frames where there are no conventional indicators of falls, say at the start of the video which depicts the subject simply walking into the room, would also be labelled as a fall. (See Figure 5)

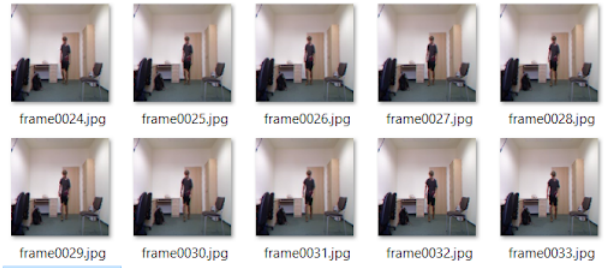


Figure 5: First 10 frames from a video in the UR Fall Dataset classified as "Fall", despite the person not having begun to fall yet.

To address this, we **re-trained the model on a lower frame rate** (view our implementation here). Given the relatively high frame rate of the UR Fall Dataset (30 FPS), and the fact that the average industry frame rate of security video cameras is 15 FPS (Segal 2021), it is reasonable to train our model on lower frame rates such that it is more suited for adaptation to CCTV cameras. In addition, we hypothesize that a lower frame rate would still be able to sufficiently capture a fall event, and we test this through our research.

As shown in Fig 6, instead of choosing consecutive frames to form a stack, the next frame chosen would be a few frames after, thus creating evenly interspersed frames. This effectively reduces the frame rate of the stack, allowing the stack to cover a larger time range and include the fall event. We then trained the model with stacks generated from a sliding window with step size = 1.

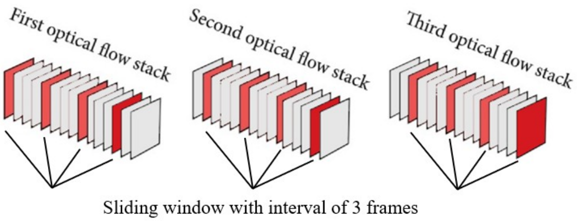

First optical flow stack
Second optical flow stack
Third optical flow stack
Sliding window with interval of 3 frames

Figure 6: Sliding window with evenly separated intervals of 3 frames.

In a real-world context, generating stacks of optical flow images, then running predictions on them in real-time, would be computationally intensive. By increasing the time between frames and thus reducing the frequency of prediction, computational requirements would also be reduced, making the system more feasible for real-world deployment.

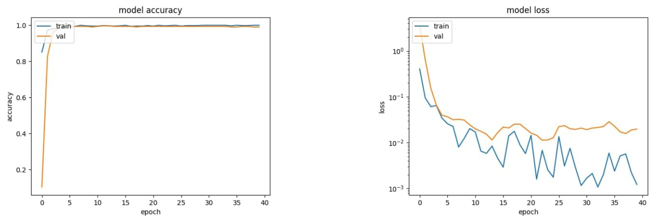


Figure 7: Accuracy and Loss of our model, against number of training epochs.

4.2 Reducing Number of Training Epochs

As observed from Figure 7, the accuracy and loss of our improved model plateaus long before the 3000 epochs Núñez et al. (2017) set. Based on this graph, we decided to drop the number of epochs from 3000 to 20. We aimed to achieve comparable performance, whilst reducing the risk of overfitting, and reducing training time. We adopted the conventional approach of taking a testing set comprising 20% of the dataset.

4.3 Tweaking VGG-16

Our team also experimented with fine-tuning of the VGG16 model. Instead of freezing the weights of the "FC6" layer, we experimented with unfreezing it and training it together with the "FC7" and "FC8" layers as well. Results are elaborated in Results and Evaluation and our implementation can be found here

Considering developments in CNN research ever since Núñez et al. (2017) released their work, a different, more modern CNN architecture could be used for the feature extractor, which could achieve better performance. For example, the Meta Pseudo Label implementation of ResNet-50. Refer to Figure 8.

Model Name	Top-1 Accuracy	Top-5 Accuracy
VGG-16	74.4%	91.9%
EfficientNet-B0	73.6%	93.2%
NoistStudent (EfficientNet-B0)	78.8%	94.5%
ResNet-50 (Meta Pseudo Label)	83.2%	96.5%

Figure 8: Table of performance of image classification models (PapersWithCode 2021)

4.4 CNN Explainer

The CNN Explainer visualizes how CNNs perceive our world. CNN explainers "start with a random noise image, and optimizes its pixels with gradient ascent so that it activates a target filter most" (Surma 2021). This highlights the patterns identified to be important by a specific CNN filter for classification of an input image. Refer to Figure 9.

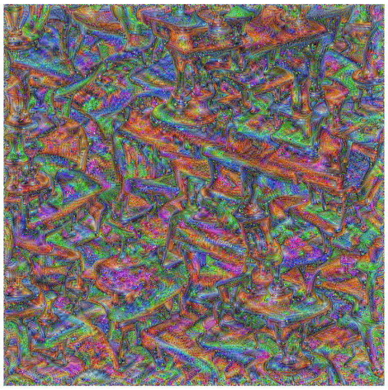


Figure 9: Input image after going through gradient ascent.

In our own testing (view our code here), we noticed that filters 325 and 419 were consistently activated for images depicting a fall, regardless of whether it was an RGB image or an optical flow image. Similarly, non fall images such as standing poses, would not have a particular filter that was distinctly activated. Below is an example:

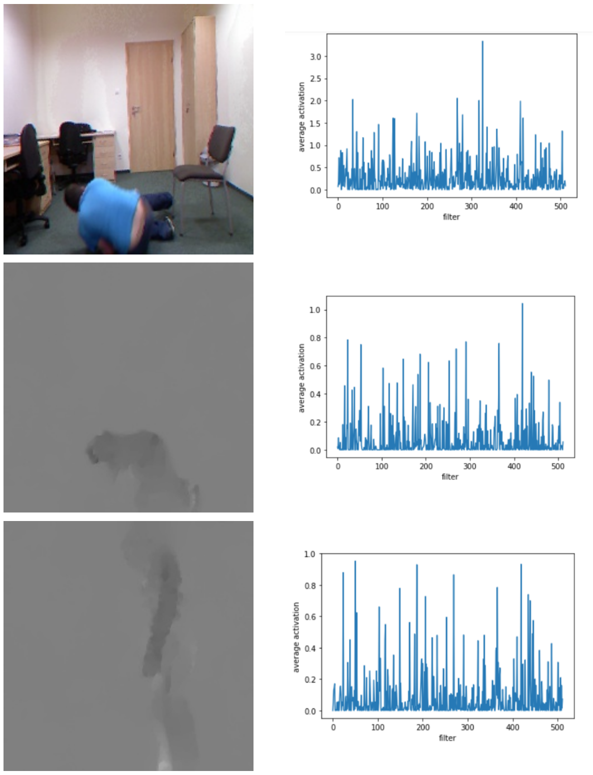


Figure 10: Input images and their corresponding CNN Filter activation graphs.

We envision that the VGG-16 CNN Explainers will shed light into the blackbox CNN model of this fall detection system (just as we have demonstrated), helping machine learning scientists to have a better understanding of the CNN and thus, improving the model. For example, analyzing why certain input videos are classified as false positives/false negatives, then tweaking training input into the model to be more robust.

4.5 Data Augmentation

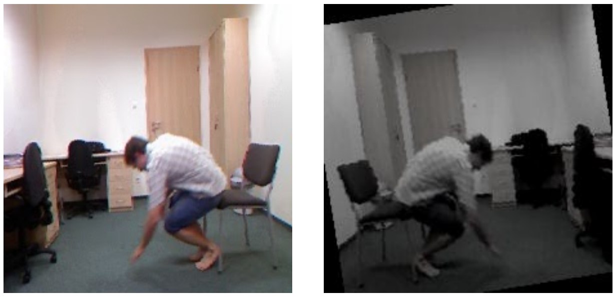


Figure 11: Original image on left, augmented image on right (darkened, flipped, rotated, blurred).

Due to the limited dataset comprising only 70 videos, we faced the **risk of overfitting our model** where our model would perform well within the UR Fall dataset but not in real world conditions. In particular, although there were variations within the falls in the video, other elements such as lighting, camera angle, position of fall and even the color of clothing remains largely the same. It would be concerning if our model could not perform in low-light conditions, such as an elderly falling while going to the washroom at night.

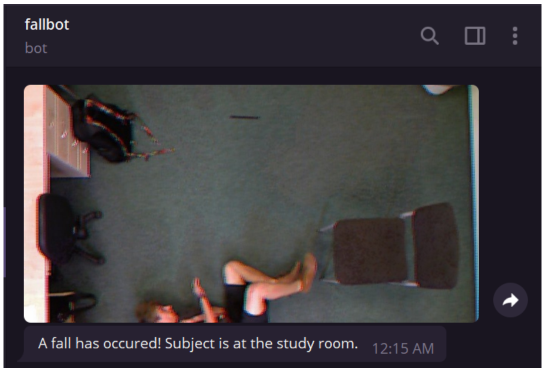
Using Python Image Library, we **experimented with changes to sharpness, brightness, contrast, saturation, color and rotation** (view our code here). The degree and type of augmentation has to be carefully managed such that training data remains realistic and mimics real world conditions. A rotation greater than 90 degrees for instance would be unrepresentative of a fall.

Implementing data augmentation increases our training data, and reduces overfitting by introducing reasonable variation. Training with this improved data leads to the model being more robust in various conditions.

4.6 Telegram Bot

We recognize that **timely alerts** of a fall to caregivers (family members and healthcare professionals) is an important part of timely response. As such, we have integrated the model to **alert caregivers of a fall through a telegram bot** that family members can subscribe to. Video data will be streamed into our hosted server where it is processed and evaluated through our model to determine if a fall has occurred. In the event of a fall, family members would then be alerted of the fall through the telegram bot. A snapshot from the camera feed, and location of the fall, will also be sent to let caregivers quickly assess the situation and respond.

In the context of fall detection for elderly, Type I errors are preferred to Type II errors, since a fall not being recognized



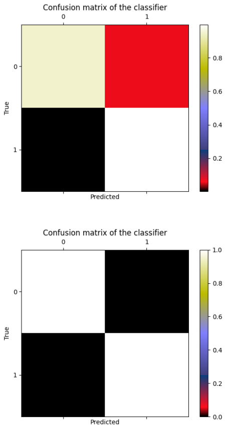
A fall has occured! Subject is at the study room. 12:15 AM

Figure 12: Telegram bot alerting caregiver of a fall.

might have serious implications. Consequently, the **prevalence of false positives** could become an annoyance for subscribers. To tackle this potential problem, we can tweak the alertion threshhold of the model such as only sending an alert after 3 consecutive positives.

5. Results and Evaluation

To evaluate the performance of our models in relation to Núñez et al. (2017) **we ran predictions using the URFD as our test set**. The method employed for preprocessing grayscale optical flow images for each model to predict, was the **same method for generating training data for that model**. (e.g. Sliding window of consecutive frames for Núñez et al.'s model (2017) vs sliding window of evenly interspersed frames for the modified models) We generated a **normalized confusion matrix for each test, and calculated the accuracy, recall, precision and F1-measure** as our performance measures. The results are as follows in Figure 13:



Núñez et al. (2017) (urfd_fold_1):
Accuracy: 99.39%
Recall: 94.67%
Precision: 99.78%
F1-measure: 97.15%

Our model (evenly interspersed frames, unfreeze FC6 layer):
Accuracy: 99.92%
Recall: 99.90%
Precision: 100%
F1-measure: 99.49%

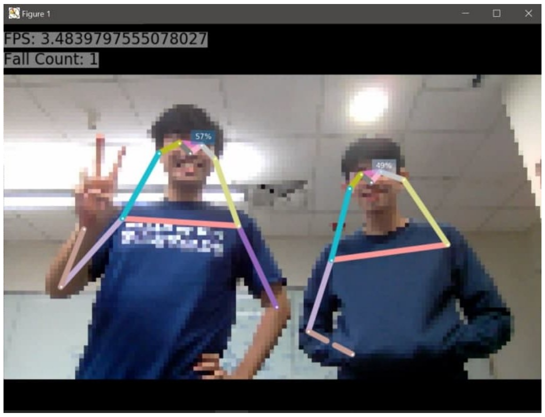
Figure 13: Model Results of Núñez et al.'s (2017) model against ours.

Overall, our approach of reducing frame rates of the optical flow stack has achieved comparable, or even slightly better, performance as compared to Núñez et al.'s (2017). Additionally, we found that unfreezing the weights of the FC6 layer during training gave a marginal improvement. Unfreezing any more preceding layers resulted in net worse performance.

Given that our data processing style is less resource intensive, our proposed approach is a positive step towards the creation of a system for fall detection in primary care facilities.

6. Limitations and Extensions

CNN model predictions are **computationally intensive** and take a substantial amount of time before returning a result. Coupled with the challenge of a constant stream of video data, it is not feasible for the model to provide real-time predictions for fall detection. We have worked towards solving this issue by altering the sliding window and reducing the amount of video data required to make an accurate decision. However the need for near immediate prediction requires us to further increase the processing speed of our model. One possible solution could be to **downsample each frame such that predictions rely on fewer features**. Alternatively, we have explored a different approach based on a threshold-based model **OpenPifPaf**. OpenPifPaf utilises human pose estimation and the relationships between key components of the human body to predict actions. We succeeded in running a fall detection model using OpenPifPaf (See Figure 14) to perform **real-time estimates and predictions**, albeit at a lower F1 score of 90.91% (Cheng 2020) while we had a F1 score of 99.95%. A possible implementation would be to feed streamed data to the more efficient OpenPifPaf model. OpenPifPaf's relatively high recall rate of 83.33% (Cheng 2020) makes it suitable to be used as a first layer of detection. When a fall is detected, we then use our model which boasts higher accuracy to predict the same set of data. Combining both a threshold-based approach and a machine-learning based approach, we could then achieve real time fall detection while maintaining comparable in prediction.



FPS: 3.4839797555078027
Fall Count: 1

Figure 14: OpenPifPaf fall detection system using pose detection.

Although there have been attempts to increase the training data through data augmentation, the dataset is still solely based on a single-person as well as a fixed background.

Moving forward we could explore other datasets which include multiple persons in one frame, or we could collect more data based on different environments. If access to actual CCTV footage of nursing homes and hospitals is possible, we could train our model using such video footages to better fit the scenario of detecting falls in hospitals and nursing homes.

As we intend to use CCTV footage as the input for our fall detection model for future use, we foresee that privacy and data security concerns regarding facial recognition could be raised. A potential way to avoid such concerns will be using real-time and automated image data anonymization tools to blur faces identified by the model so that the faces will be unidentifiable.

Team Details

- Li Keyou (A0206174L): Report, explore OpenPifPaf and OpenPose
- Maxx Chan (A0206170U): Apply, evaluate and improve Núñez et al.'s model
- Sun Hao Ting (A0206156L): Apply CNN Explainer on modified VGG-16, Telegram Bot, explore AlexNet
- Ek Chin Hui (A0205578X): Data Augmentation using Python Image Library, Telegram Bot, explore OpenPifPaf
- Tay Yu Hong (A0189568A): Apply, evaluate and improve Núñez et al.'s model
- Xu Ziyi (A0204675A): Report

References

[Adhikari, Bouchachia, and Nait-Charif 2017] Adhikari, K.; Bouchachia, H.; and Nait-Charif, H. 2017. Activity recognition for indoor fall detection using convolutional neural network. In *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, 81–84.

[Bloomfield 2019] Bloomfield, J. 2019. Singapore's ageing population and nursing: looking to the future.

[Cheng 2020] Cheng, W. L. 2020. Fall Detection using Pose Estimation.

[Chia 2021] Chia, H. K. 2021. TraceTogether saga: Timeline of key developments.

[Fleming and Brayne 2008] Fleming, J., and Brayne, C. 2008. Inability to get up after falling, subsequent time on floor, and summoning help: prospective cohort study in people over 90. *BMJ* 337(nov17 1):a2227–a2227.

[GovTech 2020] GovTech. 2020. Designing a wearable that could save lives.

[HealthHub 2021] HealthHub. 2021. Abcs of falls: consequences of falls in the elderly.

[Hermesauto 2016] Hermesauto. 2016. 30,000 more healthcare workers needed by 2020 to cater for Singapore's ageing population: Health Ministry.

[Hirschmann 2020] Hirschmann, R. 2020. Singapore: average weekly paid hours worked per employee 2019.

[Hirschmann 2021] Hirschmann, R. 2021. Singapore: old-age dependency ratio 2020.

[Karantonis et al. 2006] Karantonis, D.; Narayanan, M.; Mathie, M.; Lovell, N.; and Celler, B. 2006. Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. *IEEE Transactions on Information Technology in Biomedicine* 10(1):156–167.

[Kwolek and Kepski 2014] Kwolek, B., and Kepski, M. 2014. Human fall detection on embedded platform using depth maps and wireless accelerometer, Computer Methods and Programs in Biomedicine. 117(3):489–501.

[Mirmahboub et al. 2013] Mirmahboub, B.; Samavi, S.; Karimi, N.; and Shirani, S. 2013. Automatic Monocular System for Human Fall Detection Based on Variations in Silhouette Area. *IEEE Transactions on Biomedical Engineering* 60(2):427–436.

[Ng 2020a] Ng, D. 2020a. The loneliness of old age - and an experiment to see if Instagram can be a cure.

[Ng 2020b] Ng, M. 2020b. Singapore's first assisted living HDB flats for seniors to launch in Bukit Batok in Feb 2021 BTO exercise.

[Núñez-Marcos, Azkune, and Arganda-Carreras 2017] Núñez-Marcos, A.; Azkune, G.; and Arganda-Carreras, I. 2017. Vision-based fall detection with convolutional neural networks. *Wireless Communications and Mobile Computing* 2017.

[PapersWithCode 2021] PapersWithCode. 2021. Papers with Code - ImageNet Benchmark (Image Classification).

[Rougier et al. 2006] Rougier, C.; Meunier, J.; St-Arnaud, A.; and Rousseau, J. 2006. Monocular 3d head tracking to detect falls of elderly people. In *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, 6384–6387. New York, NY: IEEE.

[Segal 2021] Segal, Z. 2021. Average Frame Rate Video Surveillance Statistics 2021.

[Senior Care Singapore 2020] Senior Care Singapore. 2020. Elderly Automatic Fall Detection Devices Singapore | Preventing Injury.

[Surma 2021] Surma, G. 2021. CNN Explainer - Interpreting Convolutional Neural Networks (3/N).

[Wang et al. 2015] Wang, R.-d.; Zhang, Y.-l.; Dong, L.-p.; Lu, J.-w.; Zhang, Z.-q.; and He, X. 2015. Fall detection algorithm for the elderly based on human characteristic matrix and SVM. In *2015 15th International Conference on Control, Automation and Systems (ICCAS)*, 1190–1195. ISSN: 2093-7121.

[Zainal and Teoh 2018] Zainal, K., and Teoh, Z. W. 2018. Successful Ageing: Progressive Governance and Collaborative Communities.

[Zhang and Karunanithi 2016] Zhang, Q., and Karunanithi, M. 2016. Feasibility of unobstrusive ambient sensors for fall detections in home environment. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 566–569. Orlando, FL, USA: IEEE.

[Zigel, Litvak, and Gannot 2009] Zigel, Y.; Litvak, D.; and Gannot, I. 2009. A Method for Automatic Fall Detection of Elderly People Using Floor Vibrations and Sound—Proof of Concept on Human Mimicking Doll Falls. *IEEE Transactions on Biomedical Engineering* 56(12):2858–2867.