Volume 2, Issue 1, June 2022 - ISSN 2735 - 5977

ACADEMY Publishing Center

Journal of Advances in Computing and Engineering

Academy Publishing Center
Journal of Advances in Computing and Engineering (ACE) First edition 2021

Arab Academy for Science, Technology, and Maritime Transport, AASTMT Abu Kir Campus, Alexandria, EGYPT
P.O. Box: Miami 1029
Tel: (+203) 5622366/88 – EXT 1069 and (+203) 5611818
Fax: (+203) 5611818
Web Site: http://apc.aast.edu

Volume 2, Issue 1, June 2022
ISSN: 2735-5977 Print Version
ISSN: 2735-5985 Online Version

ACE
Journal of Advances in Computing and Engineering

Journal of Advances in Computing and Engineering (ACE) is a peer-reviewed, open access, interdisciplinary journal with two issues yearly. The focus of the journal is on theories, methods, and applications in computing and engineering. The journal covers all areas of computing, engineering, and technology, including interdisciplinary topics.
All submitted articles should report original, previously unpublished research results, experimental or theoretical. Articles submitted to the journal should meet these criteria and must not be under consideration for publication elsewhere. The manuscript should follow the style specified by the journal.
ACE also publishes survey and review articles in the scope. The scope of ACE includes but not limited to the following topics: Computer science theory; Algorithms; Intelligent computing; Bioinformatics; Health informatics; Deep learning; Networks; Wireless communication systems; Signal processing; Robotics; Optical design engineering; Sensors.

The ACE journal does not collect Articles Processing Charges (APCs) or submission fees, and is free of charge for authors and readers, and operates an online submission with the peer review system allowing authors to submit articles online and track their progress via its web interface.
The journal is financially supported by the Arab Academy for Science, Technology, and Maritime Transport AASTMT in order to maintain quality open-access source of research papers on Computing and Engineering.

ACE has an outstanding editorial and advisory board of eminent scientists, researchers and engineers who contribute and enrich the journal with their vast experience in different fields of interest to the journal.

Editorial Committee

Editor in Chief

Associate Editors

Editorial Board

Advisory Board

Administrative Committee

Journal Manager

Professor Yasser Gaber Dessouky, Ph.D
Professor, Dean of Scientific Research and Innovation
Arab Academy for Science, Technology & Maritime Transport, Egypt

Copy Editor

Professor Abeer Refky, Ph.D
Professor, Dean of the Institute for Language Studies
Arab Academy for Science, Technology & Maritime Transport, Egypt

Layout Editor and Proof-reader

Engineer Sara Gad,
Graphic Designer, Academy Publishing Center
Arab Academy for Science, Technology & Maritime Transport, Egypt

IT Manager

Engineer Ahmad Abdel Latif Goudah,
Web Developer and System Engineer
Arab Academy for Science, Technology & Maritime Transport, Egypt

Publishing Consultant

Dr. Mahmoud Khalifa
Consultant, Academy Publishing Center
Arab Academy for Science, Technology & Maritime Transport, Egypt

TABLE OF CONTENTS

http://dx.doi.org/10.21622/ACE.2022.02.1.001

Modern Architectures Convolutional Neural Networks in Human Activity Recognition

H. Mahmoud

Building Physics and Environmental Research Institute
Housing and Building National Research Center,
Cairo, Egypt

Email: {hanan252000@yahoo.com}

Received on, 28 December 2021 - Accepted on, 20 February 2022 - Published on, 10 May 2022

ABSTRACT

In recent years, many researchers have focused on using convolutional neural networks to perform human activity recognition as evidenced by the emergence of a number of convolutional neural network architectures such as LeNet-5, AlexNet and VGG16 and modern architectures such as ResNet, Inception V3, Inception-ResNet, MobileNetV2, NASNet and PNASNet. The main characteristic of a convolutional neural network (CNN) is its ability to extract features automatically from input images, which facilitates the processes of activity recognition and classification. Convolutional networks indeed derive more relevant and complex features with every additional layer. In addition, CNNs have achieved perfect classification on highly similar activities that were previously extremely difficult to classify. In this paper, the researcher evaluated modern convolutional neural networks in terms of their human activity recognition accuracy, and she compared the results with the state-of-the-art methods. In this research, the researcher used two public data sets, HMDB (Shooting gun, kicking, falling to the floor, and punching) and the Weizman dataset (walking, running, jumping, bending, one hand waving, two-hand waving, jumping in place, jumping jack, and skipping). The experimental results indicated that the CNN with NASNet architecture achieves the best performance of the six CNN architectures on both human activity data sets (HMDB and Weizman).

I. INTRODUCTION

The use of deep learning methods in human activity recognition has become the focus of many researchers. The strength of deep learning lies in its ability to extract features automatically in a task-dependent manner. It avoids reliance on heuristic handcrafted features and scales better, making it suitable for more complex behavior recognition tasks. Furthermore, the convolution neural network is sufficiently fast to online human activity recognition. A large number of deep learning techniques have been developed and successfully applied to recognition tasks. Szegedy from Google, Inc. proposed the Inception architecture for image classification [1]. He and Zhang from Microsoft introduced residual units in residual networks [2], and Szegedy and Ioffe combined residual connections with the Inception architecture to create the Inception V4 or Inception- ResNet architecture [3]. Subsequently, Zoph and Vasudevan from Google introduced the idea of building the architecture on a small data set and then transferring the result block to another, larger data set. They applied this idea to introduce NASNet [4]. Liu and Zoph used the idea behind NASNet and modified it to create the progressive PNASNet [5]. Finally, Sandler and Howard used the idea of deep convolution layers to build MobileNet, which can be used to make smaller models more efficient [6].

http://dx.doi.org/10.21622/ACE.2022.02.1.001

Many researchers have applied deep learning in human activity recognition based on its ability to extract features automatically in a task-dependent manner. Plotz, Hammerla, and Olivier discussed the utilization of a few feature learning methods, including deep learning in activity recognition systems [7]. Zeng et al. demonstrated an algorithm for human activity recognition using mobile sensors [8]. Yosins and Jeffclune demonstrated a method to quantify feature transferability from each layer of a neural network. They showed that transferability is negatively affected by two issues: optimization difficulties, which is caused by splitting networks in the middle of fragilely co-adapted layers and the specialization of higher layer features to the original task at the expense of performance on the target task [9]. Yang and Nguyen proposed a method to build a new deep architecture for CNNs to investigate multichannel time series data. The advantages of this method are that it performs feature extraction in a task-dependent manner and that the extracted features have discriminating power with respect to the classes of human activities [10]. Zeng and Menshoel proposed a CNN-based feature extraction approach that extracts scale-invariant and locally dependent characteristics from an acceleration time series. The CNN-based approach outperformed the previous state-of-the-art approaches [11].

Most human activity recognition techniques using CNNs are sensor-based techniques that require wearable sensors to be attached to the limbs and torso of a person. However, this solution is impractical for human activity recognition. Instead, the only useful way of recognizing human activity in public places is through surveillance videos. Sensor-based methods can be used for older people in homes to discover cases of faintness or falling but cannot be used in public places. Thus, the only available approach is to analyze surveillance videos.

There is a problem when comparing different approaches of human activity recognition because of differences in the data pre-processing operations, data sets, segmentation techniques and classification models. The contribution of this paper is that it studied the effectiveness of using CNNs for human activity recognition and found these techniques to be dependent on the quality of the images used in terms of individual or group activity. This study enables the researcher to evaluate and compare the different CNN architectures used for human activity recognition. She applies her framework to conduct comparative studies on two public data sets: HMDB and Weizman. The remainder of this paper is organized as follows: Section 1 provides an introduction. Section 2 describes the various methods. Section 3 presents the experimental results, and Section 4 concludes the paper.

II. CONVOLUTIONAL NEURAL NETWORKS

In 2012, the convolutional neural network architecture called AlexNet proved successful at a large variety of computer vision tasks such as object detection [12], segmentation [13], video classification [14], human pose estimation [15], object tracking [16] and super resolution [17]. These successes encouraged researchers to find even better-performing convolutional neural networks. In 2014, researchers utilized deeper and wider networks to improve the quality by building VGG [18] and GoogLeNet [19], which achieved high performance in the ILSVRC-2014 classification challenge. However, they found that executing classical convolutional neural networks such as AlexNet, VGG and GoogleNet requires numerous calculations; for example, AlexNet used 60 million parameters, VGG used 20 million parameters and GoogleNet employed 5 million parameters. Consequently, in recent years, researchers have focused on finding new CNN architectures to reduce the computational cost. Many new architectures have appeared, such as ResNet, Inception, Inception-ResNet, MobileNet, NASNet and PNASNet.

http://dx.doi.org/10.21622/ACE.2022.02.1.001

A) Inception

The main objectives when building optimal convolutional neural networks is to obtain high performance with low computational overhead. The inception architecture [1] has proven to be highly tunable because the researcher was able to change the number of filters in the various layers to optimize the training speed without affecting the quality of the trained network. However, the layer sizes must be tuned carefully to balance the computational burden of the various models and their sub-networks. One of the main advantages of the inception architecture is that it allows the number of units to increase at each stage without introducing an uncontrolled computational complexity. Additionally, the researcher can avoid computational difficulties by determining the width of each stage and choosing the appropriate number of stages to limit the use of computational resources. Another benefit of the inception architecture is that it intuitively aligns with the idea that visual information should be processed at various scales and then aggregated so that the next stage can abstract features from the different scales simultaneously.

The network includes different modules such as a naive module and a module to perform dimension reduction; these modules are stacked upon each other with a stride of 2 max pooling layers that halve the resolution of the grid. During training, to maintain efficient use of memory, inception modules are used only for the higher layers, while the lower layers are treated in a traditional convolutional fashion. It is preferable to use an incorporation module with dimension reduction over a naive module because the naive version suffers from one large problem: its 5 **x** 5 convolutions can be expensive on top of a convolutional layer with a large number of filters even when the number is fairly modest.

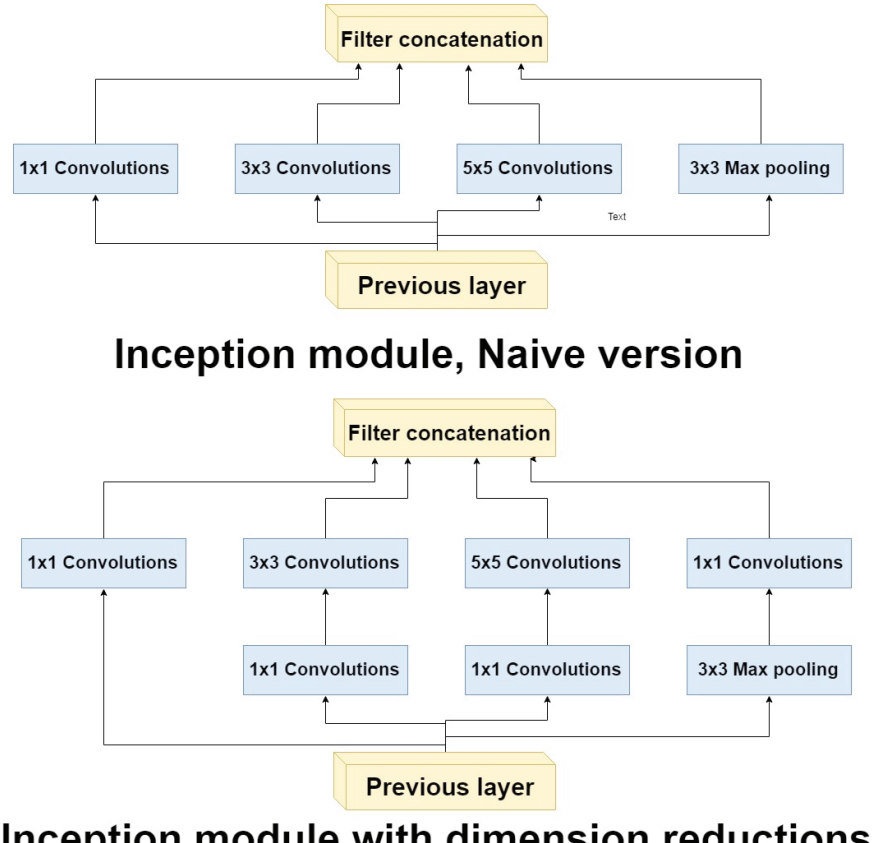


Fig. 1. Inception module Naive version and dimension reductions version

http://dx.doi.org/10.21622/ACE.2022.02.1.001

B) Inception V3

Inception V3 was introduced in 2015. It uses the inception block introduced in GoogleNet. ImageNet reduced the top-5 error rate (The Top-5 error is the portion of test images for which the correct label is not among the five labels the model considers most likely) to 5.6% (for a single model) and to 3.6% (for an ensemble model). It uses Normalization, Image distortions as open issues and RMS prop for gradient descent. It has 25 million parameters and is trained on 8 GPUs for 2 weeks. Its deep architecture consists of inception blocks. All the operations inside the inception blocks use a stride of 1 and sufficient padding to output the same spatial dimensions (W **x** H) as the feature map. Four different feature maps are concatenated on depth at the end. Inside the inception blocks, the researcher used a number of filters such as 5**x**5, 3**x**3, 1**x**1 for the convolution and pooling layers, and she added the input to the output using a single 1**x**1 convolution. Using different filter dimensions allows her to capitalize on all the features at the same time. Figure 1 shows the Inception Module Naive version.

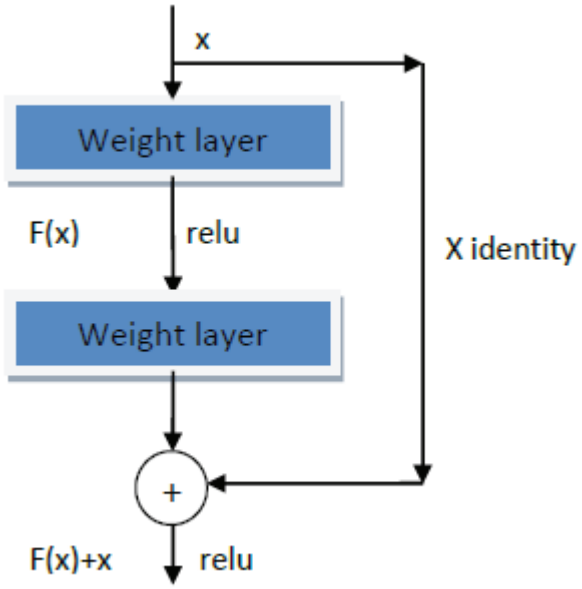


Fig. 2 Residual block

C) ResNet

Researchers designed ResNet [2] to solve the problem that as network depth increases, the accuracy first becomes saturated and then degrades rapidly. A ResNet block is either 2 layers deep (which is used in small networks such as ResNet 18 and 34) or 3 layers deep (used in larger networks such as ResNet 50, 101 and 152). Figure 2 shows a residual block. The ResNet network converges faster than its plain counterpart. ResNet 34 achieved a top-5 validation error rate of 5.71% which was better than BN-Inception and VGG. Res Net-152 achieved a top-5 validation error rate of 4.49%. An ensemble of 6 models with different depths achieved a top-5 validation score of 3.57%. ResNet 152 includes 152 layers. Figure 3 shows ResNet152 architecture, a few 7x7 convolutional layers and the rest are 3x3, batch normalization, max and average pooling layers. In total, this model has 60 million parameters and requires training on 8 GPUs for 2-3 weeks. The output channels are created by adding a small delta F(x) to the original input channels x, and F(x) is represented as a weight layer followed by ReLU activation and one weight layer. In this way, thousands of layers can be stacked, and the gradients do not vanish. Table I shows the Top-5 error rate for different ResNet architectures.

http://dx.doi.org/10.21622/ACE.2022.02.1.001

TABLE I: Top5 error for different ResNet architectures

ResNet architectures	Top5 Error
ResNet 34	5.71%
ResNet 50	5.25%
ResNet 101	4.6%
ResNet 152	4.49%

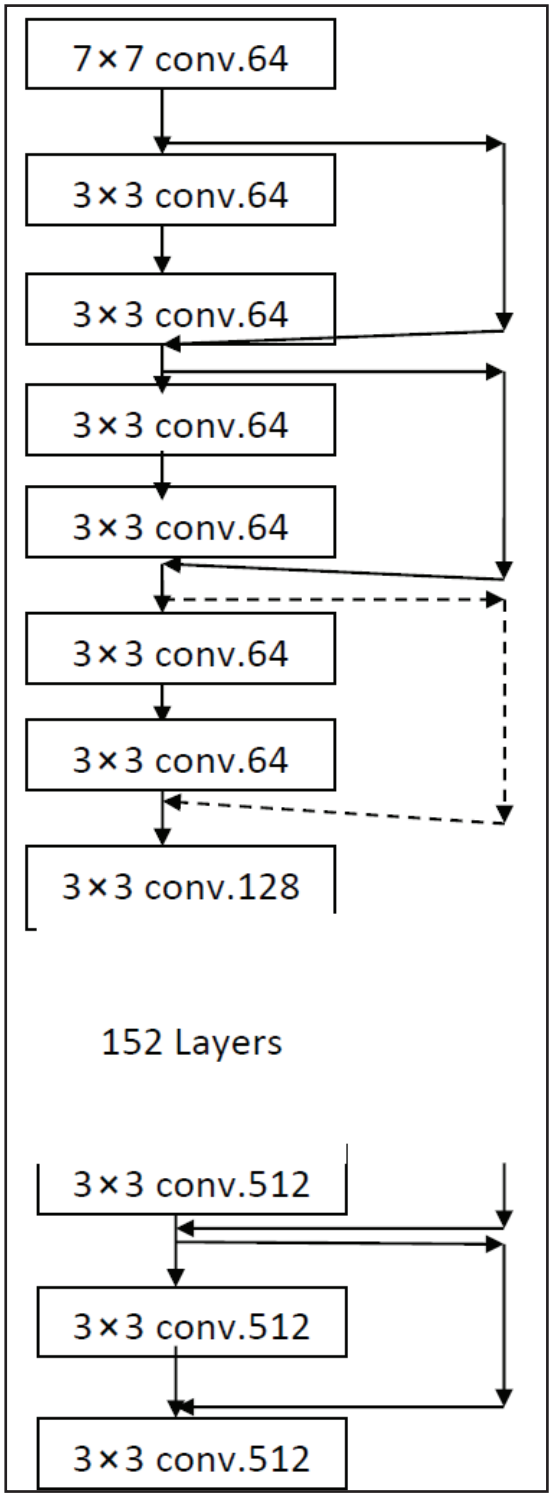


Fig. 3. ResNet152

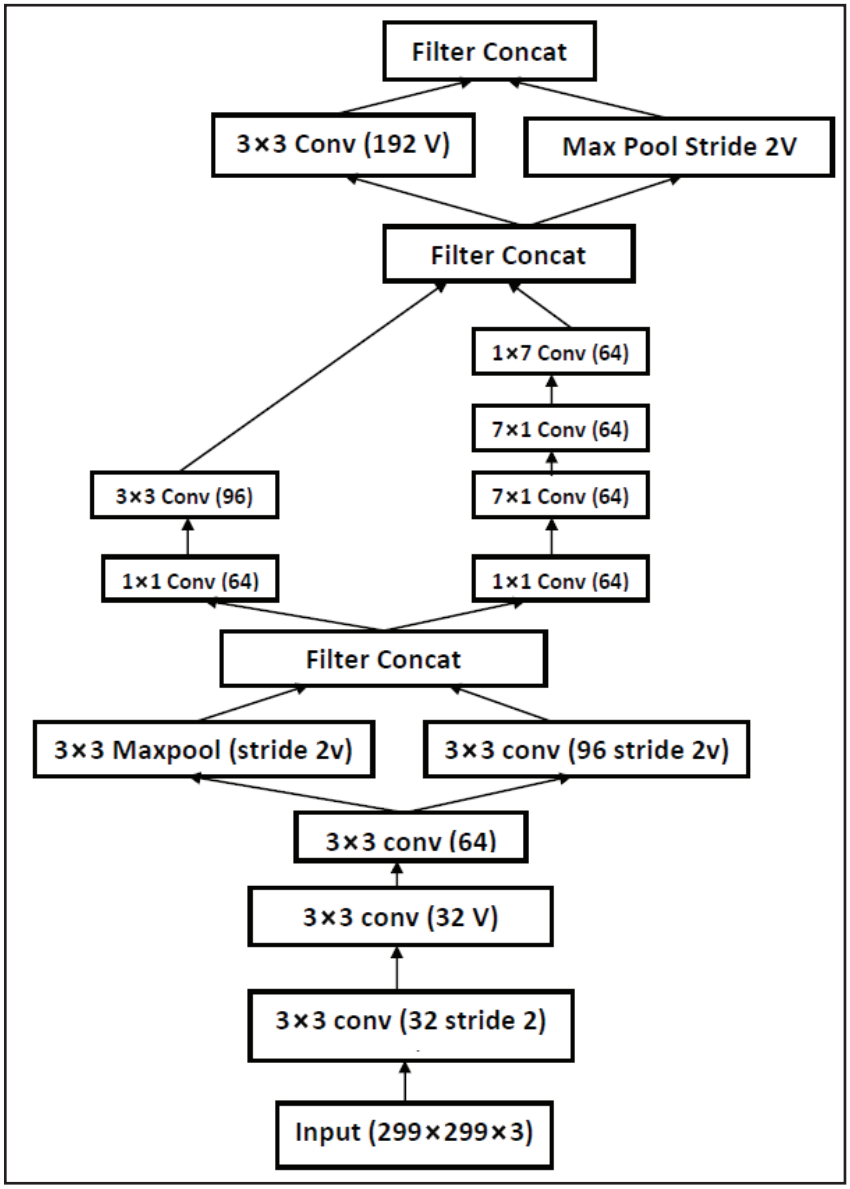


Fig. 4. Inception-ResNet architecture

http://dx.doi.org/10.21622/ACE.2022.02.1.001

D) Inception-ResNet

This architecture is the result of combining the inception architecture with residual connections [3]. By empirical evidence, the researchers found that training with residual connections accelerates the training of an inception network. Several versions of residual inceptions exist, but when the number of filters exceeded 1,000, instabilities appeared in residual variants, the network died early in the training, and after a few tens of thousands of iterations, the last layer before the average pooling started producing only zeros. Reducing the learning rate or adding extra batch normalization did not solve these problems. Scaling the residuals by factors between 0.1 and 0.3 before adding them to the accumulated layer activations seemed to stabilize the training. Figure 4 shows the Inception-ResNet architecture.

E) MobileNetV2

The architecture of MobileNetV2 [6] contains an initial fully convoluted layer with 32 filters followed by 19 residual bottleneck layers, as described in Table II. ReLU6 is adopted as the activation function, and a 3 x 3 kernel size is utilized as is standard in modern networks. Dropout and batch normalization are also used during training. The networks have a constant expansion rate except for the first layer. When the expansion rates are between 5 and 10, the performance curves are nearly identical. Larger networks achieve slightly better performance with a larger expansion rate but smaller networks perform better with slightly smaller expansion rates [6]. In MobileNetV2, the convolution is split into two layers. The first is called a depthwise convolution, and the second is called a pointwise convolution. The depthwise convolution applies a single convolutional filter per input channel to perform lightweight filtering. The pointwise convolution computes a linear combination of the input channels to buildnew features. Figure 5 shows MobileNetV2 architectures.

TABLE II

MobileNetV2: Each row describes a sequence of 1 or more identical layers, repeated n times. In the same sequence, all layers have the same number of output channels c. All layers use a stride of 1 except the first layer, which has a stride of s. The expansion factor t is always applied to the input size. All the spatial convolutions use 3 x 3 kernels. (this cannot be a title for the table; just use heading and keep the explanation to the body of the research)

Input	Operator	t	c	n	s
$224^2 \times 3$	Conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	Conv2d 1×1	-	1280	1	1
$7^2 \times 1280$	Avgpool 7×7	-	-	1	
$1 \times 1 \times 1280$	Conv2d 1×1	-	k		

http://dx.doi.org/10.21622/ACE.2022.02.1.001

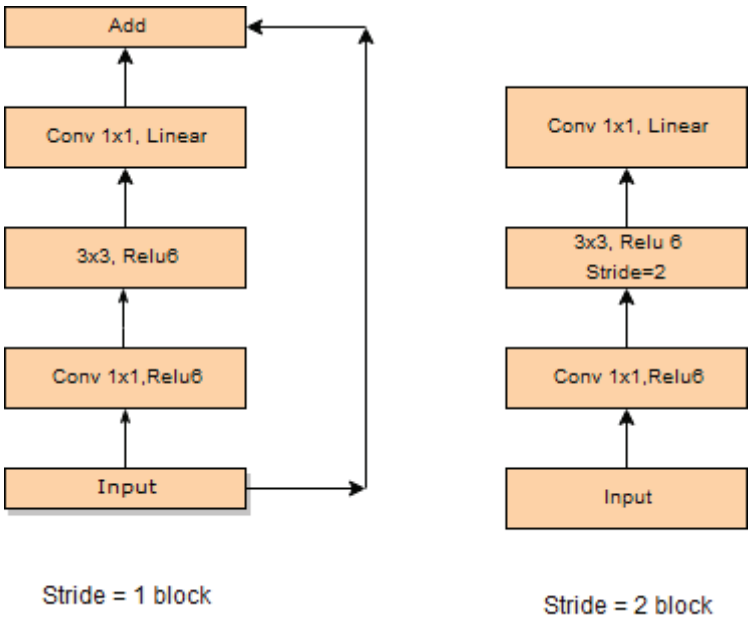


Fig 5. MobileNetV2 architectures

F) NASNet

NASNet [4] development began when Google AI introduced the AutoML project to automate the design of machine learning models. They found that AutoML can design small neural networks that perform at levels equal to those of neural networks designed by human experts. These small neural networks have achieved strong performances on small academic data sets such as CIFAR-10 (Data set consists of 60000 (32 x 32) colour images in 10 categories with 6000 images per category) and Penn Treebank. The designers also tried to apply the AutoML method to larger data sets such as ImageNet image classification and COCO object detection, which are two of the largest data sets for computer vision. However, the designers found that if naively applying AutoML directly to ImageNet required many months of training; therefore, they redesigned the search space so that AutoML can find the best layer and then repeat it many times to create a final network.

The designers performed architectural searches on CIFAR10 and transferred the best architecture to ImageNet image classification and COCO object detection. Using this approach, AutoML can find the layers that not only work best on small data sets such as CIFAR10, but also work well on large data sets such as ImageNet image classification and COCO object detection.

These two layers were combined to form the NASNet architecture. In NASNet although the general structure is predefined as a series of normal cells (convolutional cells that return a feature map of the same dimension) and reduction cells (convolutional cells that return a feature map where the height and width of the feature map are reduced by a factor of two), blocks are not defined Or cells previously by the authors instead, they are searched by the reinforcement learning search method.

NASNet achieved a prediction accuracy of 82.7% [4] on the validation set, and it performed 1.2% better than all previous models, including Inception V2, V3, Xception, Inception V4, Inception-ResNet V2 and PollyNet. NASNet can be resized to produce a group of models that achieve good accuracy at a low computational cost. Figures 6 and 7 show a schematic of the best performing reduction cell in NASNet with 4,5 blocks identified using CiFAR-10.

http://dx.doi.org/10.21622/ACE.2022.02.1.001

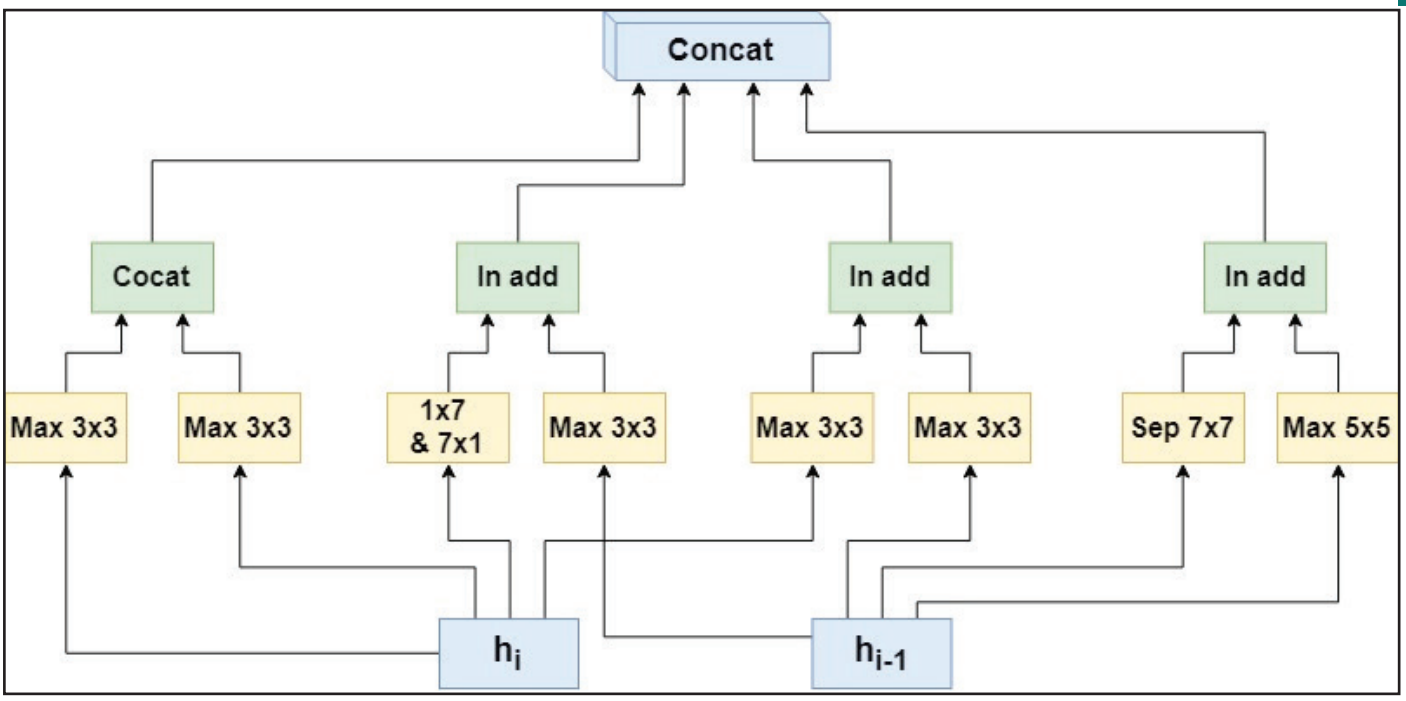


Fig. 6. Reduction cell of the NASNet architecture, 4 blocks, with CIFAR-10 identification

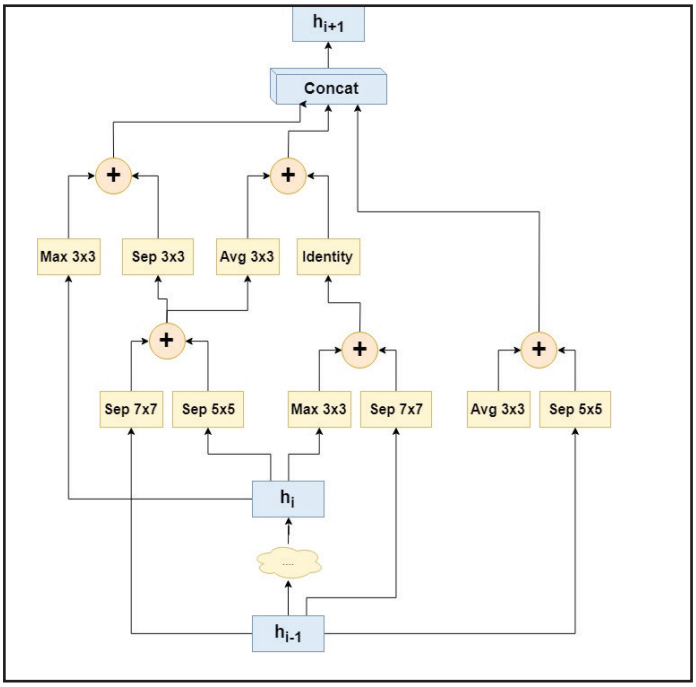


Fig. 7. Reduction cell of the NASNet architecture, 5 blocks, with CIFAR-10 identification

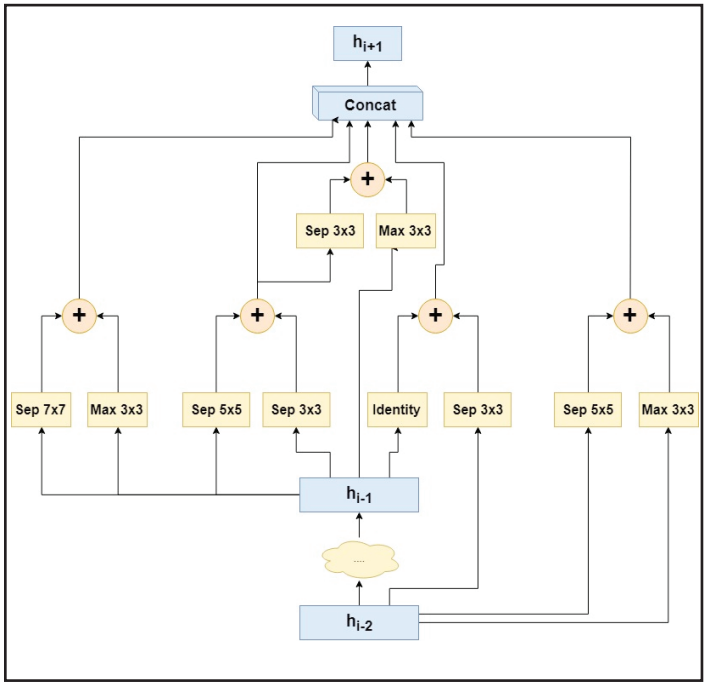


Fig. 8. Cell structure of the progressive neural architecture, with 5 blocks

G) Progressive Neural Architecture Search (PNAS)

PNAS [5] has achieved state-of-the-art classification accuracies on ImageNet and CIFAR-10. PNAS has several advantages over other techniques; for example, it can be trained faster due to its simple structures, allowing the researcher to factorize the search space into a collection of smaller search spaces and to potentially create models with many more blocks. PNAS is 5 times more efficient than is the reinforcement learning (RL) method [20] in terms of the number of models evaluated, and it is 8 times faster in terms of total computation. PNAS is a surrogate-based

http://dx.doi.org/10.21622/ACE.2022.02.1.001

search method. Surrogate-based optimization is a method that depends on learning a surrogate function that expresses a relationship between sampled models and validation errors. PNAS achieved high performance on the CIFAR-10 data set. It performs a progressive scan of the neural architecture search space. The validation errors are collected by training the selected architectures for several epochs; then, the top performing architectures are chosen at each step of the algorithm. These errors are used to train the surrogate function that predicts the validation error of subsequent architectures. The surrogate function reduces the number of architectures that actually need to be trained, thus allowing an efficient exploration of the search space. To achieve the best results requires 100 GPUS working for 2 days. PNAS is more efficient than either NAS or previous methods, which depend on up to 800 GPUs working for a month. Figure 8 shows the cell structure of PNAS.

III. EXPERIMENTAL RESULTS

In this section, the researcher presents and evaluates 286 videos from two data sets: HMDB [21] and Weizman [22]. The eighty-six videos from the Weizman data set include nine activities (bending, jumping jacks, jumping, jumping in place, running, skipping, walking, waving one hand and waving two hands), and the 200 videos from the HMDB data set include four activities (falling to the floor, punching, kicking and shooting a gun). The researcher divided the data set into two parts: 50% for training and 50% for testing where it is optimized. She evaluated the performance of six CNN network architectures, namely, ResNet, Inception V3, Inception-ResNet, MobileNetV2, NASNet and PNASNet for recognizing human activities.

For the CNNs, the ResNet block is either 2 layers deep (used in small networks such as ResNet 18 and 34) or 3 layers deep (used in larger networks such as ResNet 50, 101 and 152). She used a 3-layer deep architecture in ResNet152 and noticed that ResNet converges faster than its plain counterpart. For the Inception architectures, the researcher used Inception V3 and We also tested Inception-ResNet V2, which achieved slightly better results, and MobileNetV2. The main difference between the MobileNet architecture and a traditional CNN is that instead of a single 3×3 convolutional layer, followed by batch normalization and ReLU, MobileNets splits the convolution into two 3x3 depth-wise and point-wise convolutional layers.
In addition, the researcher also tried large NASNet and PNASNet architectures to choose the best combination of layers and connections that reach higher accuracy on human activity recognition tasks and where it is not guaranteed that a network that works well on one problem can work well on another. Because significant changes may be required to achieve accurate results, the researcher needed to select parameters such as the learning rate and initialization values carefully to avoid overfitting problems. Figure 9 shows the CNN model using a Tensor Flow diagram.

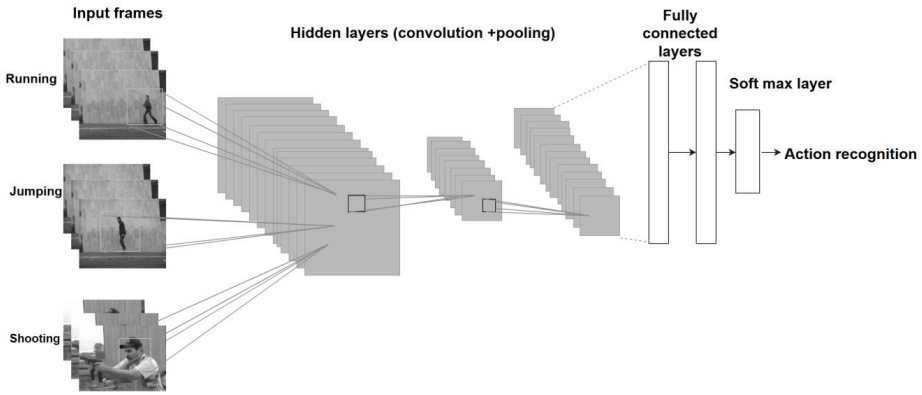


Fig. 9. Overview of the CNN model with Tensor Flow

http://dx.doi.org/10.21622/ACE.2022.02.1.001

A) Data sets Description

This section includes a complete description of each data set used. The researcher conducted her experiments on two data sets: HMDB [21] and Weizman [22]. The HMDB data set was collected from various sources, mostly movies, but a small proportion comes from public databases such as the Prelinger Archive, YouTube and Google videos. The HMDB data set contains 6,849 clips divided into 51 action categories, each containing a minimum of 101 clips. These data set include suspicious behaviours such as falling to the floor, punching, kicking and shooting a gun because the goal of this work was to recognize suspicious behaviors to prevent crimes before they happen. The Weizman data set includes 10 types of activities (walking, running, jumping, galloping sideways, bending, one hand waving, two-hand waving, jumping in place, jumping jacks and skipping). Each action is performed by nine actors. In total, the researchers selected 86 video sequences, each with a spatial resolution of 180 × 144 pixels and a frame rate of 50 frames per second.

B) Hardware and Software Setup

To perform deep learning training on training data with more than 10,000 larger images, GPU training techniques must be applied; when the equivalent training is performed on a CPU with the same data, it would take weeks to complete the training. Therefore, the researcher tried to perform training using a suitable GPU to achieve good, fast training. The NVIDIA GeForce GTX 1060 GPU (6 GB version) includes 1,280 Cuda cores, 6 GB of RAM and a memory bandwidth of 192 GB/sec, allowing training to be performed using 100 images per batch without introducing memory failures. An Intel i5-8400 CPU was used to control the entire process. The researcher used the newest Ubuntu operating system (18.04) with Python because the latter has good support for matrix libraries. She used the newest CUDA toolkit (9.2) to control the GPU-accelerated processing on our GPU. Finally, to implement the deep learning models, she used TensorFlow-GPU 1.8. TensorFlow is the brainchild of Google and intended to assist all deep learning researchers and engineers in implementing deep learning models. Additionally, TensorFlow comes with a toolkit called Tensor Board. The researcher used this tool during training to create real-time charts showing the number of steps, the training error, and the validation error, making it easier to know where overfitting may happen during the training process and to make comparisons between the various DL models. Using TensorFlow code for all the deep learning models, the training and validation data sets, and the configuration file for the training process makes it easier for researchers to make any required modifications in the models during the entire deep learning process. Figure 10 shows the hardware and software setup used for these experiments.

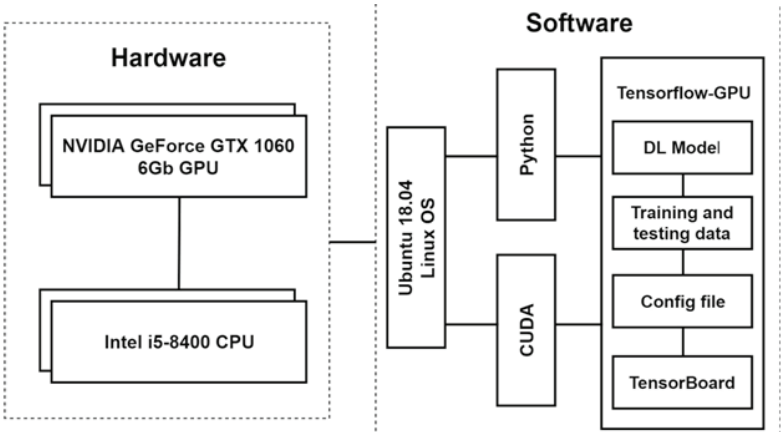


Fig. 10. Hardware and software setup

http://dx.doi.org/10.21622/ACE.2022.02.1.001

C) Results

In the first group of experiments, the researcher trained the models using multiple images of different human activities. For the CNN Inception architecture, she trained using 299 x 299 -pixel images and achieved a recognition accuracy of 86.36% on the HMDB data set and 89.9% on the Weizman data set. Additionally, she trained the CNN ResNet architecture using 299 x 299 -pixel images and used 224 x 224-pixel images for the CNN MobileNetV2 architecture, 331 x 331-pixel images for the NASNet architecture and 331 x 331 –pixel images for the PNASNet architecture. The training was performed over 30,000 steps, with 100 images per step. She found that by adjusting the input image scale, she was able to achieve smaller computational budgets and state-of-the-art performances. The best performance of the six CNN architectures for human activity recognition was achieved by the NASNet architecture, which was best on both the HMDB data set and Weizman data set.

The performance of PNASNet ranked second after the NASNet performance on the HMDB data set, but MobileNetV2 ranked second on the Weizman data set. Tables III and VI show recognition rate comparison with different approaches. The difference in performance between Weizman and HMDB data sets is due to the differences between the two data sets. The Weizman data set videos typically include only one person performing one activity in front of a static background, but the HMDB data set contains videos of group activities with variable backgrounds, which cause some occlusions. When objects are occluded by people or things, some missing measurements occur.

Therefore, the researcher preprocessed the images using a Kalman filter to counteract the missing measurements, which led to increases in recognition accuracy. Tables IV and V show the accuracy scores for human activity recognition in percentages by class. The highest accuracy percentages are defined by red rectangles.

TABLE III
COMPARISON OF CNN WITH DIFFERENT ARCHITECTURES WITH STATE- OF-THE-ART METHODS FOR WEIZMAN DATA SET

Method	Recognition accuracy
CNN+ Inception v3	89.9%
CNN+ResNet	89.3%
CNN+Inception ResNet	90.1%
CNN+MobileNet v2	90.4%
CNN+NASNet	91%
CNN+PNASNet	89.3%
Multilevel K-means[23]	98.9%
Fusing appearance and distribution information of interest points [24]	96.66%
Space time shape [25]	99.63%
Histogram of oriented rectangles and encoded with BoVW [26]	93.3%
3D SIFT + SVM classifier [27]	82.6%
Spatial- temporal words [28]	90%
Histograms of oriented 3D spatiotemporal gradients [29]	100%
mid-level motion feature [30]	100%
An Efficient Human Activity Recognition Technique [31]	100%
An information-rich Sampling Technique over Spatio-Temporal CNN [32]	95.78%

http://dx.doi.org/10.21622/ACE.2022.02.1.001

TABLE IV
ACTION RECOGNITION ACCURACIES (IN PERCENTAGES) BY CLASS ON THE WEIZMAN DATA SET

Method	Bending	Jumping jacks	Jumping	Jumping in place	Running	Skipping	Walking	Waving one hand	Waving two hands
CNN +Inception V3	98.3	89.7	87	97.8	86	86	94.7	85	85
CNN+ResNet	98.5	91	86	97.7	84	85	93.1	84.6	84.4
CNN+Inception ResNet	89.2	96.5	84	98	87	88	95	92	82
CNN+MobileNetV2	99	98.2	89	97.5	98	82.9	90	79	80
CNN+NASNet	99.5	98	92	99	90	89	89	82	81
CNN+PNASNet	99.9	90	83.4	99.5	88	81.9	90	85	86

TABLE V
ACTION RECOGNITION ACCURACIES (IN PERCENTAGES) BY CLASS ON THE HMDB DATA SET

Method	Falling to the floor	Punching	Kicking	Shooting a Gun
CNN +Inception V3	87.77	88.46	84.2	85
CNN+ResNet	88.77	87.38	89	89
CNN+Inception ResNet	81	92.31	88.33	89
CNN+MobileNetV2	85.8	88.4	88.3	90.1
CNN+NASNet	90	90.7	89.6	88
CNN+PNASNet	89.6	90.6	88.5	87.9

TABLE VI
COMPARISON OF CNN WITH DIFFERENT ARCHITECTURES WITH STATE-OF THE-ART-METHODS FOR HMDB DATA SET

Method	Recognition accuracy
CNN+Inception V3	86.36%
CNN+ResNet	88.5%
CNN+Inception ResNet	87.66%
CNN+MobileNetV2	88.15%
CNN+NASNet	89.57%
CNN+PNASNet	89.15%
STIP with HOG, HOF are encoded with various encoding methods [33]	29.22%
Improved dense trajectory [34]	57.2 %
iDT with HOG, HOF, MBHx, MBHy [35]	61.1%
Improved dense trajectory [36]	66.79 %
Spatial stream ConvNets and optical flow [37]	59.4%
Improved dense trajectory with HOG, HOF, MBHx [38]	65.1%
Trajectory-pooled deep-convolutional descriptor [39]	65.9%
STIP with HOG3D and encoded with various encoding methods [40]	30.5%
Three stream sequential deep trajectory descriptor [41]	65.2%
Human Action Recognition using Multi-Kernel Learning for Temporal Residual Network [42]	68.4%

http://dx.doi.org/10.21622/ACE.2022.02.1.001

IV. CONCLUSION

In this paper, the researcher presented a comparison between human activity recognition accuracy using convolutional neural networks of different architectures (ResNet, Inception V3, Inception- ResNet, MobileNetV2, NASNet, and PNASNet). She employed two data sets: the HMDB data set contains examples of group activities, and the Weizman data set contains examples of individual activities. The experimental results indicated that the CNN with the NASNet (Network architecture search) architecture achieves the best performance of the six CNN architectures on both the human activity data sets (HMDB and Weizman) where NasNet is certainly a more advanced technology for searching compact and efficient networks. In addition, she compared the results achieved with few of the state of the art methods.

REFERENCES

[1] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, "Rethinking the inception architecture for computer vision,". in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA 2016, pp. 2818-2826,

[2] K. He, X. Zhang, S. Ren, J. Sun, "Identity Mappings in Deep Residual Networks," in European Conference in Computer Version (ECCV), Amsterdam, 2016, pp. 630-645

[3] C. Szegedy, S. Ioffe, V. Vanhoucke, A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," in the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, California, USA, 2017, pp. 4278-4284.

[4] B. Zoph, V. Vasudevan, J. Shlens, V. Le. Quoc "Learning Transferable Architectures for Scalable Image Recognition," in Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City Utah US state, 2018, pp 8697-8710

[5] C. Liu, L. Jia Li, Barret Zoph, Maxim Neumann, et al., "Progressive Neural Architecture Search". Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 19-34.

[6] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City UT, USA 2018, pp4510-4520

[7] T. Plotz, Y. Hammerla, P. Olivier, "Feature learning for activity recognition in ubiquitous computing," in International Joint Conference on Artificial Intelligence (IJCAI), Barcelona, 2011, pp.1729-1734.

[8] M. Zeng, L. T, Nguyen, "Convolutional neural networks for human activity recognition using mobile sensors," in Mobile Computing, Applications and Services (MobiCASE) 6th International Conference, Austin, USA, 2014, pp.197–205.

[9] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, "How transferable are features in deep neural networks?" in Advances in Neural Information Processing Systems 27 (NIPS '14), NIPS Foundation, Montreal, Canada, 2014, pp. 3320-3328.

http://dx.doi.org/10.21622/ACE.2022.02.1.001

[10] J. Yang, M. Nguyen, "Deep Convolutional Neural Networks On Multichannel Time Series For Human Activity Recognition," in 24th International Joint Conference on Artificial Intelligence (IJCAI), Argentina ,2015, pp. 3995-4001.

[11] M. Zeng, O. Mengshoel, "Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors," in 6th International Conference on Mobile Computing, Applications and Services (MobiCASE-16), Austin Texas US, 2014, pp197-205.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik,Rich "Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus ,Ohio 2014, pp1-21.

[13] J. Long, E. Shelhamer, T. Darrell, "Fully convolutional networks for semantic segmentation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA2015, pp 3431–3440.

[14] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Fei- Fei, "Large-scale video classification with convolutional neural networks," in Computer Vision and Pattern Recognition (CVPR) IEEE, Columbus, Ohio, 2014, pp 1725–1732.

[15] A. Toshev, C. Szegedy, "Deeppose, Human pose estimation via deep neural networks" in Computer Vision and Pattern Recognition (CVPR) IEEE, Columbus, Ohio 2014, pp 1653–1660.

[16] N. Wang, D.Yeung, "Learning a deep compact image representation for visual tracking," in Advances in Neural Information Processing Systems Conference, US, 2013, pp 809–817.

[17] C. Dong, C. Loy, K. He, X. Tang, "Learning a deep convolutional network for image super-resolution," in Computer Vision–ECCV Conference, Zurich, Switzerland, 2014, pp 184–199.

[18] K. Simonyan, A. Zisserman, "Very deep convolutional networks for large-scale image recognition", ICLR Conference, San Diego, May 2014, pp. 1-14.

[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, et al., "Going deeper with convolutions," in IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, June 2015, pp 1-9.

[20] B. Zoph, V. Vasudevan, J.Shlen,Le, Q.V, "Learning Transferable architectures for scalable image recognition," in IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1-14.

[21] HMDB Data set@http://serre-lab.clps.brown.edu/resource/ hmdb-a-large-human-motion-database.

[22] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri, "Actions as space-time shapes, IEEE Trans.Pattern Anal. Machine Intell, vol. 29, issue 12, pp. 2247-2253. Dec. 2007.

[23] M. Elshourbagy, E. Hemayed, M. Fayek, "Enhanced bag of words using k-means

http://dx.doi.org/10.21622/ACE.2022.02.1.001

for human activity recognition,"Egyptian Informatics Journal, vol.17, issue2 ,PP 227-237, July (2016).

[24] M. Bregonzio, T. Xiang, S. Gong, "Fusing appearance and distribution information of interest points for action recognition". Pattern Recognition Journal (Elsevier), vol 45, issue 3, March 2012, pp 1220–1234.

[25] M. Blank, L. Gorelick, E. Shechtman, M. Irani, R. Basri, "Actions as space-time shapes," in Tenth IEEE International Conference on Computer Vision (ICCV'05), Beijing, China, 2005, pp. 1395–1402.

[26] N. Ikizler, P. Duygulu, "Human action recognition using distribution of oriented rectangular patches," in Human Motion–Understanding Modelling Capture and Animation Conference, Rio de Janeiro, Brazil, 2007, pp. 271- 284.

[27] P. Scovanner, A. Ali, M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in Proceedings of the 15th ACM international conference on Multimedia, Augsburg, Germany, 2007, pp.3 57–360.

[28] J.Carlos, H.Wang, L.Fei-Fei, "Unsupervised learning of human action categories using spatial- temporal words," Int J comput vision, vol. 79, pp.299-318, March 2008

[29] A. Klaser, M. Marszalek, C. Schmid, "A spatio-temporal descriptor based on 3d-gradients," in Proceedings of the British Machine Vision Conference Leeds, United Kingdom, 2008, pp. 271–275.

[30] A.Fathi, G.Mori, "Action recognition by learning mid-level motion features," in IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, 2008, pp. 1–8.

[31] A. Khelalef, Fakhreddine, N. Benoudjit, "An Efficient Human Activity Recognition Technique," HAL open science Journal, vol 29, issue 4, pp 702-715 , June 2020.

[32] SH.Shabbeer Basha, "An information-rich sampling Technique over spatio-temporal CNN for classification of human actions in video," Computer vision and pattern recognition Journal, arxiv:2002.02100v2, 7Feb (2020).

[33] X.Wang, L.Wang, Y.Qiao, "A comparative study of encoding, pooling and normalization methods for action recognition," in Computer Vision, Asian Conference on Computer Vision, Sydney, Australia., 2012, pp. 572–585.

[34] H.Wang, C.Schmid, "Action recognition with improved trajectories," in IEEE International Conference on Computer Vision, Sydney, Australia, 2013, pp. 3551–3558.

[35] X.Peng, L.Wang, X.Wang, Y.Qiao, " Bag of Visual Words and Fusion Methods for Action Recognition, Comprehensive Study and Good Practice: Elsevier Vol. 150 pp. 109–125, September 2016.

[36] X.Peng, C.Zou, Y.Qiao, Q.Peng, "Action recognition with stacked fisher vectors," in Computer Vision, Asian Conference on Computer Vision, ECCV, Zurich, 2014, pp. 581-595.

[37] K.Simonyan, A.Zisserman, "Two-stream convolutional networks for action

http://dx.doi.org/10.21622/ACE.2022.02.1.001

recognition in videos," in Advances in Neural Information Processing Systems Conference, Montreal, Quebec, Canada. 2014, pp. 568–576.

[38] Z.Lan, M.Lin, X.Li, A.G.Hauptmann, B.Raj, "Beyond gaussian pyramid: multi-skip feature stacking for action recognition," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA., 2015, pp. 204–212.

[39] L.Wang, Y.Qiao, X.Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2015, pp. 4305–4314.

[40] X.Zhen, L.Shao, "Action recognition via spatio-temporal local features: a comprehensive study," Image and Vision Computing, vol. 50, pp. 1–13, June 2016.

[41] Y.Shi, Y.Tian, Y.Wang, T.Huang, "Sequential deep trajectory descriptor for action recognition with three-stream CNN," IEEE Transactions on Multimedia, vol 19 issue7, pp. 1510–1520, July 2017.

[42] S.Nazir, M.Haroon, S.Velastin, "Human Action Recognition using Multi- Kernel Learning for Temporal Residual Network," in 14th International Joint Conference Vision, Imaging and Computer Graphics Theory and Applications, Prague Czech Republic, 2019, pp.420-426

http://dx.doi.org/10.21622/ACE.2022.02.1.017

Customer Active Power Consumption Prediction for the Next Day Based on Historical Profile

Ahmad A. Goudah[1], Mohmed El-Habrouk[2], Dieter Schramm[3] and Yasser G. Dessouky[1]

[1]Arab Academy for Science, Technology and Maritime Transport,
[2]Department of Electrical Engineering, Faculty of Engineering, Alexandria University, Egypt,
[3]Institute of Mechatronics University Duisburg-Essen D-47057 Duisburg, Germany,

Email: {ahmad.goudah@aast.edu, eepgmme1@yahoo.co.uk, dieter.schramm@uni-due.de, ygd@aast.edu}

Received on, 12 January 2022 - Accepted on, 07 May 2022 - Published on, 19 June 2022

ABSTRACT

Energy consumption prediction application is one of the most important fields that is artificially controlled with Artificial Intelligence technologies to maintain accuracy for electricity market costs reduction. This work presents a way to build and apply a model to each costumer in residential buildings. This model is built by using Long Short Term Memory (LSTM) networks to address a demonstration of time-series prediction problem and Deep Learning to take into consideration the historical consumption of customers and hourly load profiles in order to predict future consumption. Using this model, the most probable sequence of a certain industrial customer's consumption levels for a coming day is predicted. In the case of residential customers, determining the particular period of the prediction in terms of either a year or a month would be helpful and more accurate due to changes in consumption according to the changes in temperature and weather conditions in general. Both of them are used together in this research work to make a wide or narrow prediction window.

A test data set for a set of customers is used. Consumption readings for any customer in the test data set applying LSTM model are varying between minimum and maximum values of active power consumption. These values are always alternating during the day according to customer consumption behavior. This consumption variation leads to leveling all readings to be determined in a finite set and deterministic values. These levels could be then used in building the prediction model. Levels of consumption's are modeling states in the transition matrix. Twenty-five readings are recorded per day on each hour and cover leap years extra ones. Emission matrix is built using twenty five values numbered from one to twenty five and represent the observations. Calculating probabilities of being in each level (node) is also covered. Logistic Regression Algorithm is used to determine the most probable nodes for the next 25 hours in case of residential or industrial customers.

Index Terms—Smart Grids, Load Forecasting, Consumption Prediction, Long Short Term Memory (LSTM), Logistic Regression Algorithm, Load Profile, Electrical Consumption.

http://dx.doi.org/10.21622/ACE.2022.02.1.017

I. INTRODUCTION

SMART grids, which are robust and efficient power grids, are going to become the grids of the future. Smart grids, basically, consist of a two-way flow of electricity and information as the U.S. Department of Energy (DOE) suggests. This is used to monitor customer electrical consumption behavior on the grid [1]. At the same time, Advanced Metering Infrastructure (AMI) is collecting real time data during the day as consumption information is considered as an important and essential data logger resource. Millions of smart meters are assembled worldwide in Smart Grids. Electrical industry is deregulated globally due to the increase in demand consumption [2].

Detailed daily customer consumption individual data is obtained by AMI in order to achieve a better figure about customer behaviors. It is worthwhile to note here that customers' consumption behavior patterns are completely different even if they are from the same category (residential, commercial, industrial, etc.) [3]. Work in this paper depends on data provided by a Spain company coordinated with Oviedo university in Spain. This data set is for the hourly consumption profile (Active and reactive power) of more than 3,000,000 clients in a period of three years as shown in figure 1.

In this work, extracting customer consumption patterns from his load profiling is done by using a Hidden Markov Model to predict future consumption for that customer.

- **Dia:** The date on which the energy usage occurred, in the following format: YYYY/MM/DD
- **H1, H2,..., H25:** The time at which the energy usage occurred. There are 25 entries due to the time change when the clocks are set back one hour on the last Sunday of October each year, which leads to a total of 25 hours on this day.
- **ACTIVA_H1, ACTIVA_H2,..., ACTIVA_H25:** Active energy consumed per hour as measured in kWh (kilowatt-hour). This is the useful energy that the customers absorb from the grid and transform into work and/or heat at home.
- **REACTIVA_H1, REACTIVA_H2,..., REACTIVA_H25:** Reactive energy consumed per hour in VArh (volt-ampere-hours). This is a supplementary usage that the customers cannot take advantage of. Currently reactive energy consumption is not billed, although it could be charged for in future as a way of improving the energy efficiency of homes.
- **DE_MUNICIP:** The municipal district to which the customer belongs. Geographical reference.
- **FECHA_ALTA_STRO:** Date on which the customer's service was activated with the following format: QXAAA
- **TARGET_TENENCIA_CUPS:** Probability that the municipal district in question is already equipped with a natural gas distribution network (which does not imply that the customer had contracted natural gas service).
- **IDENTIFICADOR:** The unique, customer reference number which allows for segmentation of usage per customer.
- **CNAE:** (National Classification of Economic Activities) This value indicates whether the customer is domestic (T1) or not (T2).
- **PRODUCTO:** Tariff / electrical product that the customer has contracted; there are up to 120 products.
- **MERCADO:** This value indicates whether the customer has a regulated tariff (M1) or a free market tariff (M2)
  - Regulated Tariff: the price of the electricity is regulated periodically by the corresponding authority.
  - Free market: the price of the electricity is freely agreed upon by the provider and the customer.

Fig. 1. Electrical company data structure model

http://dx.doi.org/10.21622/ACE.2022.02.1.017

However, customer consumption prediction is useful for determining the future electrical needs for city, governorate and country now and in the coming years [1]. This information about what is predicted for the future is translated directly into profits and gradually affect companies marketing strategies [4]. Such systems are improving the overall efficiency of the electricity network and a better consumption behavior is achieved. This informative system leads to a good understanding for organizations to accurately target customer behavior modifications and act accordingly [2], [4]. Different customer behaviors are useful for electrical companies to offer some incentives to change household electrical behavior to a much proper way [2]. Electrical Demand Market now is ready for expectations released from customer consumption prediction to lead a successful energy utilization [5]. In this paper, Hidden Markov Model (HMM) is used to model customer historical data for the average of 300 days consequently recorded. Then Logistic Regression Algorithm is applied to determine the most probable consumption levels sequence during the day. The same process is applied to customers-base of different users to apply the idea and verify that it is a valid prediction model.

A. System Block Diagram

Household consumer Demand Response (DR) term concerns with three types of processes applied to customer's energy consumption. These types are categorize, predict and modify customer's energy consumption. It is an important tool for improving a utility's economic and energy efficiency, reducing emissions, and integrating renewable devices [2].

Paper [2] presents a shape-based approach and is mainly based on Dynamic Time Warping DTW. Hidden patterns of regular consumer behavior are observed and reflected by an optimal alignment between energy consumption patterns using DTW. Under DTW distance, two valuable benefits were achieved. Firstly, 50% reduction in the number of representative groups of electrical household consumers. Secondly, measured prediction accuracy is improved. Extendedly, determining which device is used in any particular hour from consumption curve analysis. Used devices and their fundamental structures are the main base to make classification for household customers' consumptions. DTW method is comparing and classifying household load curves. DTW, K- means and Gaussian based E&M algorithms are mentioned in this paper as different clustering algorithms. Three topics are discussed in [2]. First, clustering with DTW.

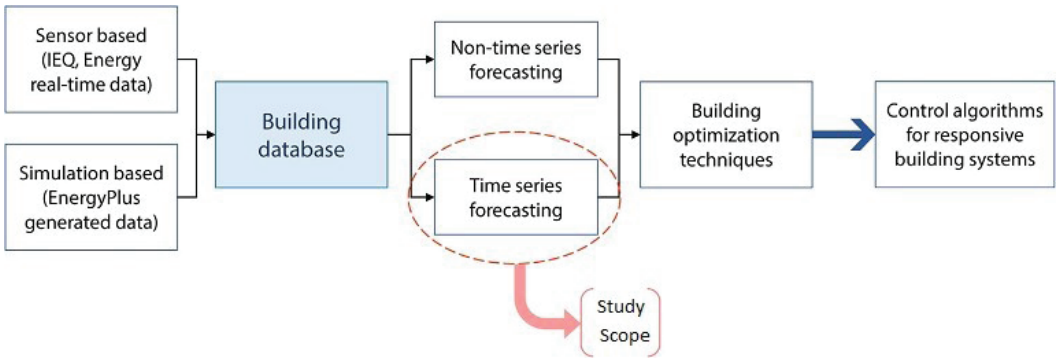


Fig. 2. Modified generalized block diagram [1]

http://dx.doi.org/10.21622/ACE.2022.02.1.017

Second, applying Markov Model based method to predict one day ahead under a shape-based measure. Finally, a new method is introduced to estimate used devices in household consumptions. Most forecasting techniques model a relationship between aggregated load demand and driver variables such as calendar effect, weather effect and lagged load demand (e.g. demand at previous hours or at the same hours of previous days). [3] Presents the definition of the clustering as "the data mining technique where similar data are placed into related or homogeneous groups without advanced knowledge of the groups' definitions".

Unlabeled data set of objects are distributed among groups according to the maximum similarity that has to be in the same group. Exploratory data analysis process is using clustering to form data into similar groups for summary generation as a preprocessing step. Time-series data analysis is used to do many tasks in many fields for different purposes like: subsequence matching, anomaly detection, motif discovery, indexing, clustering, classification, visualization, segmentation, identifying patterns, trend analysis, summarization, and forecasting. Time-series clustering is a special type of clustering which is working on continuous, real-valued items and is dynamic because values and observations are changed as a function of time and are considered as temporal data which is huge in size and highly dimensioned in structure. Time-series clustering is a challenging problem because of:

- data normally is larger than memory size and is stored in disks which leads to decrease in the speed of the clustering process.
- data are often high dimensional which makes handling these data difficult for many clustering algorithms.
- the similarity measures that are used to make the clusters depend on "whole sequence matching" where whole lengths of time-series are considered during distance calculation which is complicated, because time-series data are naturally noisy and include outliers and shifts which makes similarity measure select is a challenge in itself.

[4] used a big data approach to make load forecasting. The multiple linear regression model is introduced to find the optimal number of lagged hourly temperature and moving average temperature needed in a regression model that gives the best mean DTWE across 150 households in validation set. [5] presents Semi-parametric additive model to make day ahead (short term) and year ahead (middle term) load forecasting from data collected every ten minutes by ERDF at 2260 substations located at the frontier between the high voltage grid and the distribution network in France. In [6], Support Vector Machine (SVM) algorithm is used on load forecasting.

B. Modeling and Methodology

Time-series modeling scheme is better than Temperature that should not be considered in during the period in which the temperature does not vary much. In [7], Artificial Neural Network (ANN) hourly short-term electric load forecasting system is used as shown in I-A. It is known as ANNSTLF (Artificial Neural Network Short Term Load Forecaster) and is the most widely used ANN-based load forecaster in the USA and, possibly, the world. ANNSTLF can model the effect of two major weather variables on the load, temperature, and relative humidity.

http://dx.doi.org/10.21622/ACE.2022.02.1.017

The ANNSTLF package also includes an ANN hourly temperature forecaster and an hourly relative humidity forecaster. In [1], Time series electrical consumption forecasting models have strong contribution in optimization and planning fields in both buildings and compounds. Machine learning and statistical algorithms are used to predict and forecast future consumption based on historical readings and it is proven to be accurate and fast way than other methods. Historical data of energy consumption is analyzed often with various variables like weather, temperature, humidity, consumption season and other environmental conditions.

Combined methods of machine learning are much effective than using just one method. Reducing costs and carbon emissions in efficient buildings are commonly discussed topics related to energy, economic and environmental aspects. Energy consumption is affected by many factors like weather conditions, occupancy schedule, thermal properties of building materials, complex interactions of the energy systems like HVAC and lighting, etc. Complex relations between these factors are very difficult to simulate or using in simulation programs. Depending on data derived from consumption directly is a good scenario to study these effects of multi-dimensional problem. Algorithms used for that are depending on history readings and past patterns of electrical consumption. Time series algorithms are used as machine learning data-driven algorithms which depend on past customer electrical consumption. Electrical consumption forecasting is very important because it gives very clear boundaries for future consumption of a day, week, month and year and how this will affect economically.

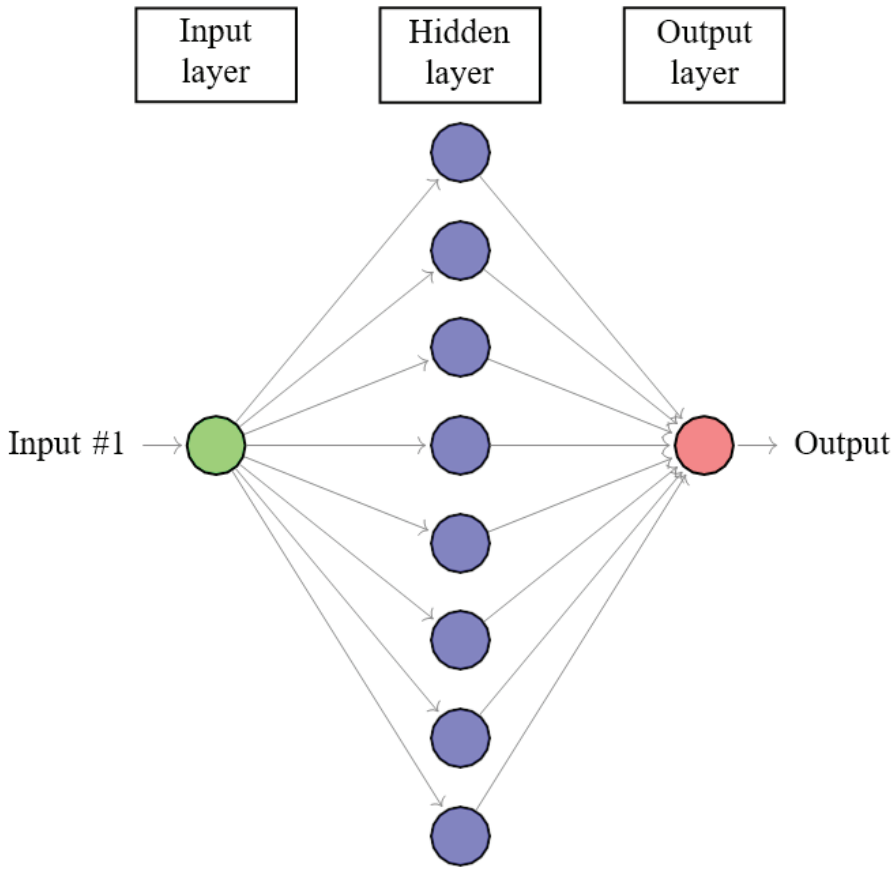


Fig. 3. Traditional shallow layer perceptron Artificial Neural Network Model with 8 neurons in hidden layer

http://dx.doi.org/10.21622/ACE.2022.02.1.017

In [8], time series forecasting model learned for customer behavior to give building managers a full view of usage patterns in spot and by comparing these consummations by recorded ones in the same conditions, a good view appears. Time series and non time series forecasting models could be used to predict occupancy and other operational factors. Forecasting and energy optimization are highly dependant. They are fully integrated because of the always need for information to calculate the best consummation configuration and future vision from electrical consumption predictors. Machine learning algorithms are used here widely to answer questions about optimal consumption behavior which takes in consideration patterns of consumption and predictions for future consumption too.

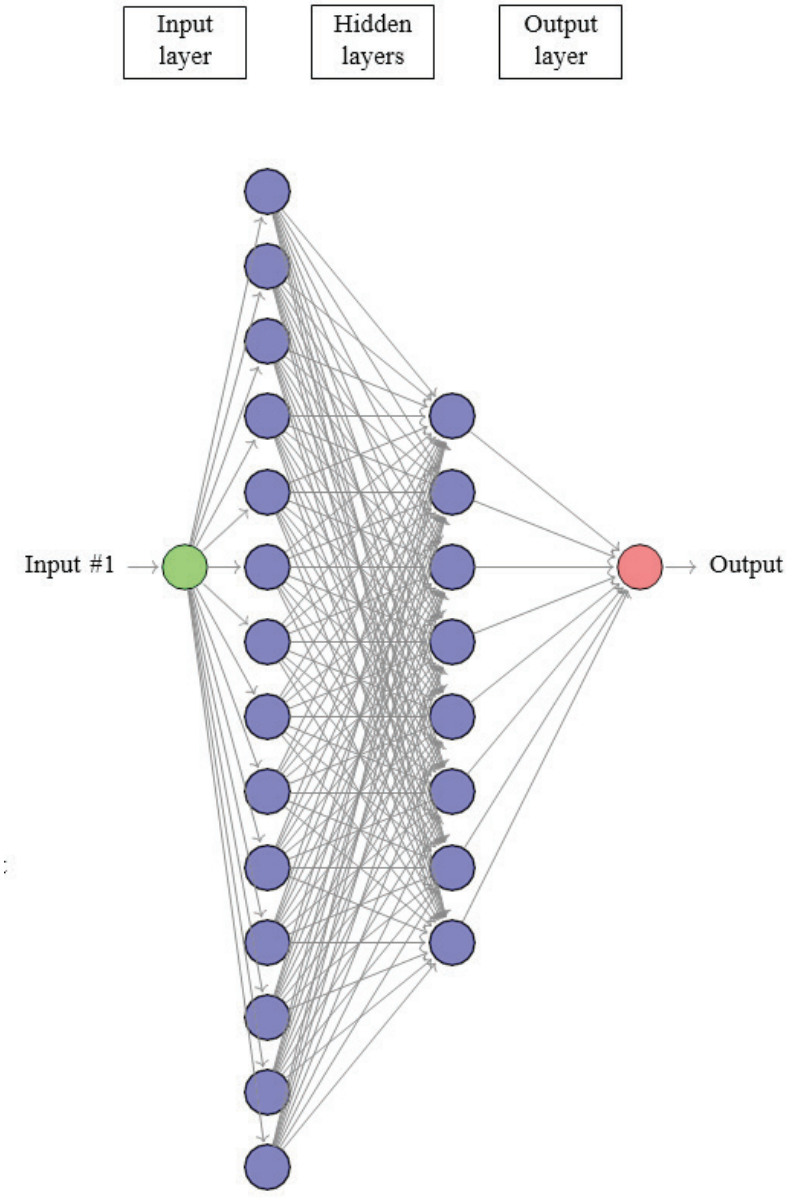


Fig. 4. Deep perceptron Artificial Neural Networks with two hidden layers with 14 neurons in first hidden layer and 8 neurons in second hidden layer

http://dx.doi.org/10.21622/ACE.2022.02.1.017

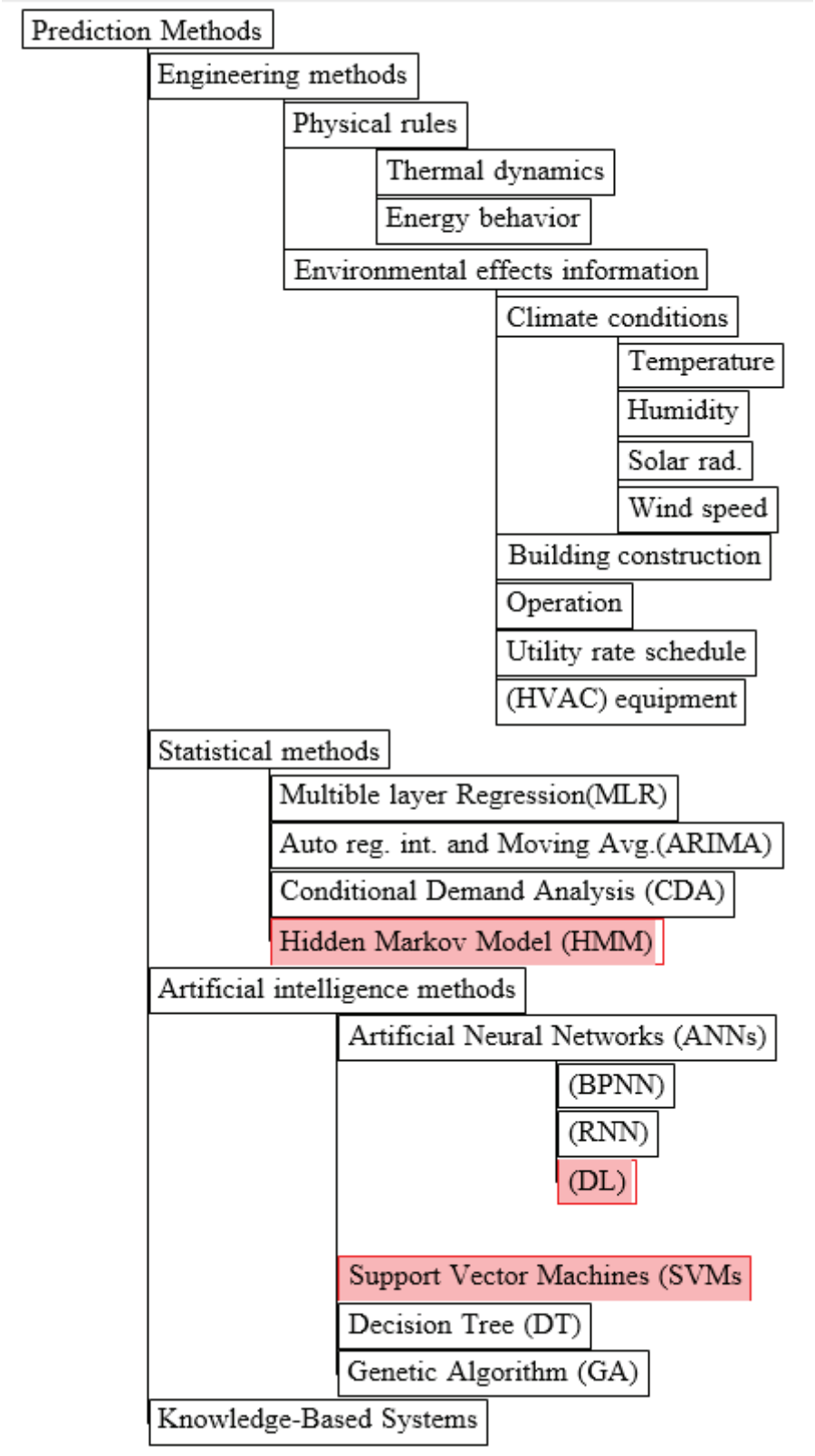


Fig. 5. Customer consumption prediction categories and methods with selected algorithms

http://dx.doi.org/10.21622/ACE.2022.02.1.017

In [9], There are many factors that highly affect electrical consumption like temperature, building construction and thermal properties of material which are used in building construction. Due to these complicated factors, prediction of household electrical consumption is very difficult.

As I-A shows, there are mainly three modeling categories or methods which are used to make customer prediction vision for building consumption. These categories are:

- **Engineering methods:** use physical rules such that thermal dynamics and energy behavior of the electrical consumption in buildings with environmental effects information,

such as:
– External climate conditions like:
* Temperature
*Humidity
* Solar radiation
* Wind speed
– Building construction
– Operation
– Utility rate schedule
– Heating, Ventilation and Air-Conditioning (HVAC) equipment.

- **Statistical methods like:**

– In [10], statistical regression algorithms such as:
***Multible layer Regression (MLR)** is used and by predicting a day ahead, load forecasting is obtained in this model [11] [12]. Regression coefficients were found out with the help of method of least square estimation [13] [14]. Load in electrical power system is dependent on temperature, due point and seasons and also load has correlation to the previous load consumption (Historical data) [1] [15]. Then, the input variables are temperature, due point, load of prior day, hours, and load of prior week [16] [17]. To validate the model or check the accuracy of the model mean absolute percentage error is used. Using day ahead forecasted data weekly forecast is also obtained [11] [18]. Load forecasting mean forecasting average load in KW or total load in KWh for periods or blocks of 15 minutes, 30 minutes, 1 hour, day, week, month or a year for daily forecast, weekly forecast, monthly forecast or yearly [19] [20]. There are many factors which influence the accuracy of load forecasting like weather variables, holidays, festivals or events, tariff structures, available historical data, time of the year, day of the week and hour of the day. Weather variable includes temperature, humidity, rain and wind. Temperature and humidity has a considerable effect on power consumption because as the temperature rises people turn on air conditioners and if temperature is low air heaters will be turned on, which increases electricity demand. If there is a celebration, the electricity demand will rise due to lightnings [21] [10].

***Auto regressive, Integrated and Moving Average (ARIMA)** is one of the most popular including time series analyses. ARIMA method and regression analysis, is one of traditional methods that based on mathematical calculations, but ARIMA is one of the most recent approaches including Kalman filtering, BoxeJenkins models, and state space models [1] [22] [20].

http://dx.doi.org/10.21622/ACE.2022.02.1.017

Hybrid model called SVRARIMA that sums up the Support Vector Regression (SVR) and ARIMA models are used to have better forecasting performance than individual model [23] [24] [25]. All of these methods can achieve electric load forecasting but cannot receive the desired prediction accuracy because of their limitations. For example, linear regression depends on historical data and cannot solve non-linear problems. Autoregressive moving average models give the result taking only into account the past and current data points while ignoring other influential elements. The grey forecasting model can only effectively solve the problem with exponential growth trends [26] [27]. Literature shows that time series analysis techniques are neither scalable to higher dimension nor are effective in highly volatile data [28] [29] [11]. For this reason time series methods such as regression models, ARIMA models, GARCH and hybrid models such as combination of ARIMA and GARCH using wavelet transform are not considered for short term forecasting [30] [31].

ARIMA processes are well suited to express the stochastic nature of the load time series [1] [14]. Modeling of multiple seasonal cycles as well as introducing exogenous variables is not a problem in ARIMA [6] [32]. The disadvantage of ARIMA models is that they are able to represent only linear relationships between variables [33] [34]. The difficulty in using ARIMA is the problem of order selection which is considered to be subjective [35] [36]. To simplify the forecasting problem, the time series is often decomposed into a trend, seasonal components and an irregular component [37] [38]. These components, showing less complexity than the original series, are modeled independently [3] [39]. The ARIMA parameters were estimated for each forecasting task (i.e. the forecast of system load at time t of the day d) using time series fragments immediately preceding the forecasted day [40] [41]. Typical days in these fragments were replaced with the days from the previous weeks [42] [43].

Due to using short time series fragments for parameter estimation (much shorter than the annual period) and due to time series decomposition into n series [1] [44], it is not required to take into account the annual and daily seasonalities in the models [10] [45] [16]. In such case, the number of parameters is much smaller and they are easier to be estimated compared to models with triple seasonality [46] [47]. This eliminates the daily seasonality and simplifies the forecasting problem [48] [49].To estimate parameters of ARIMA the stepwise procedures for traversing the model spaces implemented in the forecast environment for statistical computing [50] [15].The conventional forecasting model like ARIMA works significantly worse than the best Neural Networks models [51] [52], ARIMA and ES are optimized on the time series fragments directly preceding the forecasted fragment [53] [13]. The classical statistical model like ARIMA is the simplest among tested methods. It has only one parameter to estimate. Such a model is easy to optimize and has good generalization properties. Its learning and optimization procedures are extremely fast [54] [12] [55].

***Conditional Demand Analysis (CDA)** is a method to model the residential end-use energy consumption at the national level [56]. There are several studies where CDA was used to model energy consumption at the regional level; however the CDA method had not been used to model residential energy consumption at the national level [9] [10]. The prediction performance and the ability to characterize

http://dx.doi.org/10.21622/ACE.2022.02.1.017

the residential end-use energy consumption of the CDA model are compared with those of a neural network (NN) and an engineering based model developed earlier. The comparison of the predictions of the models indicates that CDA is capable of accurately predicting the energy consumption in the residential sector as well as the other two models. The effects of socio-economic factors are estimated using the NN and the CDA models, where possible. Due to the limited number of variables the CDA model can accommodate, its capability to evaluate these effects is found to be lower than the NN model. [57].

***Hidden Markov Models (HMM)** are useful in time series data analysis, however, its application in building energy sector is not so much investigated [2]. HMM-based procedure is used for identification of individual household appliances from collective energy consumption data [3]. However, this requires HMM models to be trained for individual appliance energy consumption profile [1]. An HMM-based modeling scheme is presented in for energy level prediction in wireless sensor networks nodes. They consider node energy level as a stochastic variable having values within certain fixed range forming hidden state for their HMM model whereas nodes energy consumption is treated as observed state [18] [58] [59].

- **Artificial intelligence methods like:**

– Artificial Neural Networks (ANNs) algorithms such as:

***Back Propagation Neural Network (BPNN)** The most common type of ANN used in classification of remote sensing imagery is the MLP (multi-layer perceptron) networks based on the BP (back-propagation) learning algorithm [1]. This type of network is called BPNN (back propagation neural network) [13] [1] [22]. Studies on the performance of back propagation networks are still ongoing. The back propagation learning algorithm is the best algorithm among the Multi-layer perceptron algorithms [34] [56] [15]. The processing speed and classification accuracy often depend on the design and implementation of the network. In the literature, different researchers introduced different network architectures for different applications. Are there any common standards among neural network researchers to guide the design and implementation of BPNNs for remote sensing image classification? This question is still open [60] [61] [10]. Choosing the number of layers and the number of neurons in each layer is essential. The performance of the feed-forward BPNNs is affected significantly by the network layout [29] [27] [24]. The selection of this layout and its related parameters is referred to as the art of network design. Unfortunately, there is no clear answer for the layout of the neural network for a particular application; however, there are some general rules which have been collected over time and followed by researchers [20] [21] [31].

Recurrent Neural Network (RNN) and Multilayer Perceptron's (MLPs) algorithms are very efficient for time series forcasting processes. Figure I-A shows the Neural Network Model used in MLP algorithm used in this study. There are many advantages of using such algorithm such as:

- Robust to Noise
- In input data and in the mapping function. Support learning and prediction in the presence of missing values [21].
- Nonlinear

http://dx.doi.org/10.21622/ACE.2022.02.1.017

- Do not make strong assumptions about the mapping function.
- Readily learn linear and nonlinear relationships [10] [62].
- Multivariate Inputs
- Input features can be specified.
- Providing direct support for multivariate forecasting [33] [63].
- Multi-step Forecasts
- Output values can be specified.
- Providing multi-step and multivariate forecasting [12] [51].

C. Results and Accuracy

In this study, the used model depends on Feed Forward Neural Networks for Time Series Forecasting. Firstly, transforming Data for Time Series calculation is done. Then sliding window transformation technique is applied too. LSTM algorithm is applied here in time series forecasting as follows:

*It is defined as the number of input time steps as three via the input dim argument on the first hidden layer.

*Efficient Adam version of stochastic gradient descent and optimizes the mean squared error ('MSE') loss function is used as shown in figures (5) and (6).

* Fitted model can be used to make a prediction

Implementing these steps using Python, NumPy and Keras for deep learning is strait forward and the results are clear in figures: I-C for Results of Naive Time Series Predictions with Neural Network and I-C for Results of Window Method for Time Series Predictions with Neural Networks respectively. From these different runs, it becomes clear that frame of the prediction problem is done Error was not significantly reduced in window method compared to that of the previous section with Naïve one. The window size and the network architecture were not tuned as more runs and modifications should be used to reach tuned output of prediction system. In this work, it is utilized to use window size of ten successive days in a row. After applying Discrete Cosine Transform DCT to the results of the prediction, Validation Error reduced to be 0.9813% for all runs. This step is mandatory after using the LSTM to reduce validation error. LSTM model obtained a higher performance than the other applied models as shown in Table 1 and due to the small sample of different target variables, it is difficult to generalize if DL models may help when applied at such levels. The experiments demonstrated that the LSTM model had the best performance under each statistical indicator.

– **Support Vector Machines (SVMs)** are one of the most probability used algorithms for datasets with high number of dimensions with small and nonlinear samples. on the other hand, these algorithms are not applicable for big datasets and they have a little big time complexity of O(N3) [57] [64] [11]. However, combining SVM with other AI models may improve forecasting accuracy [33] [65]. There is still a long way to go in order to make the methods applicable, and future work should focus on reducing the computational costs and memory while maintaining accuracy before on-line practical applications [12] [54]. SVM is a very specific technique characterized by usage of kernels, absence of local minima, sparseness of the solution and capacity control obtained by acting on the margin, or on number of support vectors [66]. The capacity of the system is controlled by parameters that do

not depend on the dimensionality of the feature space. The non-linear function is leaned by linear learning machine which maps inputs into high dimensional kernel induced feature space [13] [67]. SVM is motivated to find and optimize the generalization bounds given for regression [22] [34]. They relied on defining the so called epsilon intensive loss function that ignores errors, which are situated within the certain distance of the true value [64] [14]. Using two stages forecast engine incorporating linear regression; dynamic programming; and support vector machine (SVM) [63] [2]. Fixed size least squares support vector machines (LS-SVM) using an autoregressive exogenous nonlinear autoregressive

```
# create and fit Multilayer Perceptron model
model = Sequential()
model.add(Dense(8, input_dim=look_back, activation='relu'))
model.add(Dense(1))
model.compile(loss='mean_squared_error', optimizer='adam')
model.fit(trainX, trainY, epochs=200, batch_size=2, verbose=2)
  Train Score: 49280.91 MSE (221.99 RMSE)
  Test Score: 13341.68 MSE (115.51 RMSE)
```

```
plt.plot(dataset)
plt.plot(trainPredictPlot)
plt.plot(testPredictPlot)
plt.show()
```

Train Error: 4.19%
Test Error: 4.42%

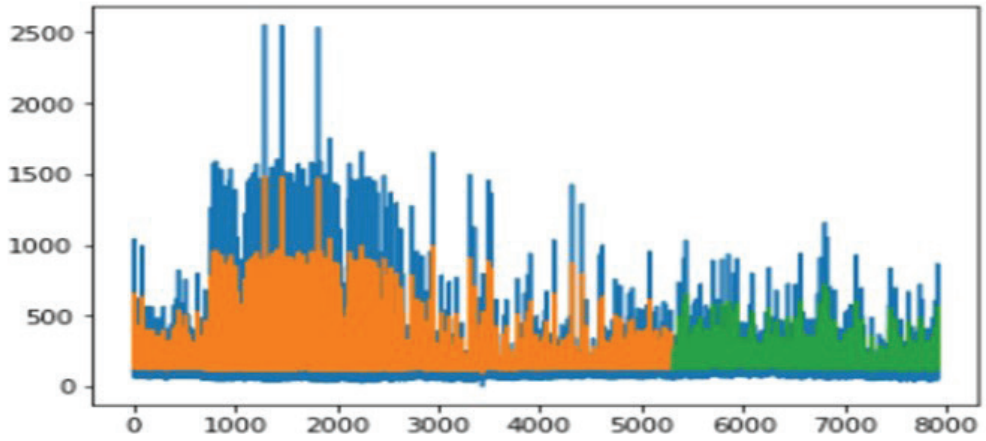


Fig. 6. Results of Naive Time Series Predictions with Neural Network

http://dx.doi.org/10.21622/ACE.2022.02.1.017


```
# create and fit Multilayer Perceptron model
model = Sequential()
model.add(Dense(14, input_dim=look_back, activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(1))
model.compile(loss='mean_squared_error', optimizer='adam')
model.fit(trainX, trainY, epochs=400, batch_size=2, verbose=2)
    ...: plt.plot(dataset)
    ...: plt.plot(trainPredictPlot)
    ...: plt.plot(testPredictPlot)
    ...: plt.show()
Train Score: 46773.37 MSE (216.27 RMSE)
Test Score: 16315.15 MSE (127.73 RMSE)
```

Train Error: 4.077%
Test Error: 4.888%

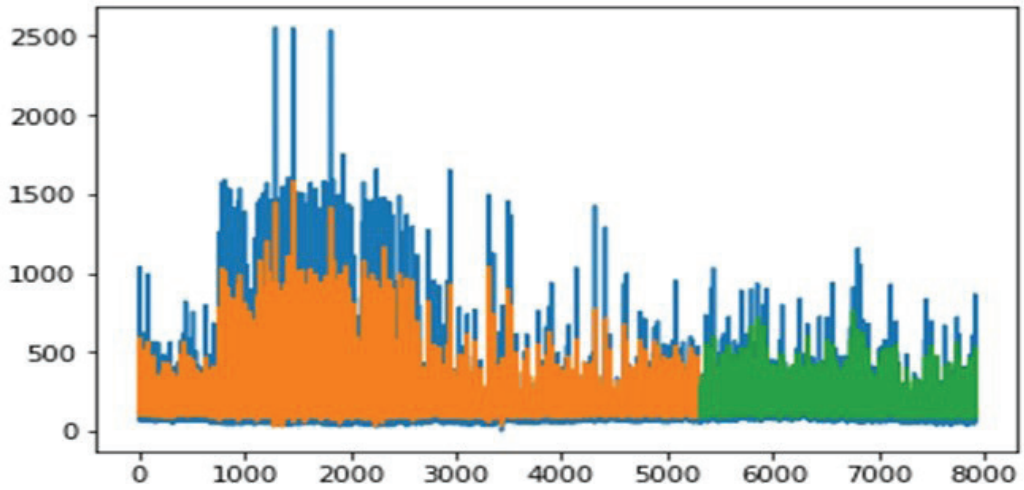


Fig. 7. Results of Window Method for Time Series Predictions with Neural Networks

exogenous structure, could also be implemented to outperform the linear model [68] [1] [56] [15]. Poor computational scalability is one of the disadvantages when using SVM's as like as using other methods such as neural networks and other computational intelligence techniques [61] [59] [16] [45]. The state of the art experimental results showed that SVM's methods are considered to be strong and have a non-linear learning capabilities for electric load forecasting which are combined with the empirical mode decomposition method and auto regression one [10] [39] [36]. The neural network easily falls into the local minimum because of the restriction on generalization ability and cannot make full use of information from selecting sample with a small sample size [28] [69].

Compared with a traditional neural network, the support vector machine (SVM) can overcome these drawbacks to improve forecasting performance. As a kernel based method, SVM employs the learning principle with structural risk minimization (SRM) to increase its generalization capability in the training process, generating better forecasts [37] [70] [60]. Because of the attractive feature and empirical performance, SVM has become one of the most promising and popular forecasting methods [11] [19] [40] [47]. Therefore, SVM is used as the forecasting method and the parameters of SVM have an important influence on the accuracy of prediction [46] [43] [71]. An alternative to solve the problem is by using heuristic optimization algorithms for parameter selection that they are

http://dx.doi.org/10.21622/ACE.2022.02.1.017

prone to be more efficient and robust than a traditional optimization algorithm, e.g., grid search algorithm [9] [50] [49] [72]. Therefore, the Cuckoo search (CS) algorithm, a heuristic optimization algorithm that has powerful ability to search for an optimal solution, is used to determine parameters of SVM [36] [11] [25] [73] [58]. In addition, Singular Spectrum Analysis (SSA), a powerful technique in time series analysis that was used for electric load forecasting, is employed to remove the high frequency components of the noisy load series in order to improve the forecasting performance of the SVM model, producing the CS- SSA-SVM model [25] [73] [62]. The combination of the data preprocessing-based technique, kernel-based method and heuristic optimization algorithm. The performance of the hybrid model is validated by forecasting the short-term electric load [24] [23].

Integrating merits of individual algorithms to enhance prediction accuracy and based on the signal reprocessing technique, the hybrid methods can robustly map input space into feature space to tackle the complex nonlinear problems [58] [21] [74]. The powerful signal processing technique is applied to decomposition and reconstruction of the original electric load data; the analysis process from embedding to diagonal averaging performs identification and extraction of different characteristics to alleviate negative effects from the noisy signal [31] [20] [53]. The powerful global search capacity of the CS algorithm is employed to serve for optimal intelligent selection of model parameters, overcoming limitations of artificially selected parameters [35] [29] [75]. Apply the hybrid model to the constructed electric load series to show its superiority compared with other benchmark models [44] [41] [6].

– **Decision Tree (DT)** have been used literally to build energy prediction system and to provide reliable promising prediction results during the past two decades [10] [72]. However, one of the major disadvantages of single prediction approach is the instability issue within each learning algorithm [10] [19]. Learning algorithms such as ANN and decision tree are unstable learners which may introduce significant variation in the output value due to some small changes made in the input data [64] [11]. This instability issue could impede these algorithms from implementation in real-time, on-the-field applications as some energy efficiency measurements rely on the reliability of the prediction [12] [76], for example, an unstable learner may lead to high false alarm ratio for building system fault detection. To overcome the limitation of instability as well as to improve the prediction accuracy, the concept of ensemble learning has been recently introduced by researchers to solve both classification and regression problems [13] [67].

To improve prediction accuracy and by using data obtained from meteorological systems and building-level occupancy and meters, the prediction performance of conventional decision tree method is improved i.e., Classification and Regression Tree (CART) by introducing bagging technique [64] [14] [34]. However, it should be noted that the feature importance results may not accurately interpret the relationship between each input feature and the output variable [77] [66]. Some of the input variables used generally, are highly correlated with each other, for example, the outdoor temperature and solar radiation variable. The decision tree may only use one of them for tree growing and put away the rest because correlated variables share the same impurity [56] [15] [60]. the feature importance of the used

http://dx.doi.org/10.21622/ACE.2022.02.1.017

variable will be high while those of the unused variables will decrease significantly. For example, the feature importance of outdoor temperature is 2.0 in module [47] [42] [78], while its correlated variable solar radiation only has feature importance of 0.5 [48] [16]. The low value of feature importance should not be taken as evidence that the variable is not strongly related to the output variable [45] [23] [21].

– **Genetic Algorithm (GA)** it is used to combine the prediction of each base model and output the final results as the prediction of the ensemble model [10] [11] [12]. Literature preliminary results showed that the proposed ensemble model provided higher prediction accuracy than the typical single model [54] [53] [14]. Recently, gene expression programming (GEP) was proposed as a new function model mining algorithm. Compared with traditional genetic algorithms (GA) and Genetic Programming (GP), GEP has advantages in terms of convergence speed and ability to solve complex problems [1] [22]. At present, research on GEP focused on symbolic regression, function finding, combinatorial optimization and prediction. In symbolic regression and function mining, an improved GEP algorithm named SGEP is proposed, which is especially suitable for dealing with symbolic regression problems [34] [79]. On the other hand, hybrid models for load forecasting algorithm included combination of genetic algorithm and ant colony optimization for feature selection and multi-layer perceptron for hourly load prediction are commonly used [79] [7].

• **Knowledge-Based Systems** Knowledge-based filtering is emerging as an important field which uses knowledge about users and products to pursue a knowledge-based approach to generating recommendations, reasoning about what products meet the user's requirements [11] [67] [14]. Gradual incorporation of different types of information (e.g., explicit ratings, social relations, user contents, locations, use trends, knowledge-based information) has forced forecasting system to use hybrid approaches. Once the memory-based, social and location-aware methods and algorithms are consolidated, the evolution of system demonstrates a clear trend toward combining existing collaborative methods [1] [34] [77]. The major reason behind using commercialized knowledge-based systems (KBS), is the lack of a strict validation step in their life cycle. There is a widespread agreement that KBS cannot be designed in a linear fashion. This is due to the typical problems they have to resolve; these will require them either to adapt traditional techniques of software development or to use new techniques relevant to artificial intelligence (AI) systems [41] [47] [46].

A significant problem in the development of Knowledge-Based Systems (KBS) is its verification step. By using Machine Learning techniques to progressively improve the quality of expert system Knowledge Bases and by coping with two major KB anomalies: incompleteness and incorrectness [9] [72] [78]. In agreement with the current tendency, KBs considered in our approach are expressed in different formalisms. Results obtained with two different learning algorithms, confirm the hypothesis that integrating machine learning techniques in the verification step of a Knowledge-Based System life cycle, is a promising approach [48] [48] [45]. Knowledge based systems (KBS) or expert systems emulate the human expert behavior in a certain knowledge area [10] [3] [32]. They constitute aid systems to take decisions in different areas such as educational strategic selection,

http://dx.doi.org/10.21622/ACE.2022.02.1.017

environmental variables control [74], neonatology fans configuration, agreement in judicial process or the attended generation of activity maps of software development projects. Knowledge based systems to aid decision taking is a particular knowledge based system [6] [31] [19].

Although, hybrid approaches are used by combining multiple methods to enhance prediction performance. In [10], for classification in data-driven models, algorithms that are usually considered are:

- **K-means Clustering Techniques – Fuzzy K-means Clustering**
  To improve the accuracy of short-term electric load forecasting for individual users, a short-term power load forecasting model based on K-means [3] [69]. By analyzing the users' electricity consumption features, K-means is applied to group users into two clusters. [2] [10].

  – Fuzzy C-means Clustering
  For users with strong correlation at adjacent moments, local similar data are filtered out with the help of improved Fuzzy C-Mean clustering (FCM), integrating the load value of the adjacent moments into new input features [80] [55]. For users with weak correlation at adjacent moments, the local similar daily data are utilized as features [60] [40]. Finally, the feather vectors are used as input data for BP Neural Network, which is utilized to forecast the short-term load [34] [15].

- **Self-organizing map (SOM)** Among different neural network classifiers, Self Organizing Map (SOM) is one of the most effective methods. SOM has two valuable features, namely, pattern recognition and pattern complementarity [1] [22] [2]. These two SOM properties are exploited for short-term load classification and for forecasting, respectively [70] [41]. SOM is using an iterative active learning technique based on self-organizing map (SOM) neural network and support vector machine (SVM) classifier [47] [50] [3]. The technique exploits the properties of SVM classifier and that of SOM neural network to identify uncertain and diverse samples to be included in the training set [6] [36] [29]. It selects uncertain samples from low density regions of the feature space by exploiting the topological properties of SOM [27] [69] [62].

In addition to the extensive use of classification methods for different applications in electrical engineering, classification methods particularly have been used for load management and load forecasting [21] [81] [67]. Doing forecasting for the short-term (the next day) load, while used system focuses only on the method of classifying daily electric load and the classification of loads with the same behavior for the time period selected (e.g. one week, one month, or one year, or any desired time interval from one day to several years) [10]. Although none of the outputs of the SOM classifier is directly used for the prediction purpose, the results of the SOM network classification process can be used as an effective way for sampling the appropriate training patterns for training and forecasting 24-hour active power consumption of the nationwide electricity. This can be used in a separate SOM network. According to the outcomes of the classifier, every Normal day of a week has its specific load consumption curve and holidays have distinguished ones [82][83].

- **Hierarchical clustering** Hierarchical time series prediction plays an

http://dx.doi.org/10.21622/ACE.2022.02.1.017

important role in various applications, such as retail in business, electricity supply and environmental protection [11] [67]. In the electricity power supply and management, the power supply and consumption is usually organized in a hierarchical structure according to administrative divisions [2] [56] [60]. In big data applications, hierarchical time series prediction is an important element of decision-making and concerns the inherent aggregation consistency, which is maintained by reconciliation methods [47] [71]. By using hierarchical forecasting approach, the k means clustering [55] based multiple alternative clustering strategy is employed to cluster the time series with different cluster number k to obtain a large number of time series clusters [10] [3] [11]. It brings more chances to build good aggregate hierarchies as the input of our hierarchical prediction model. Furthermore, instead of dealing with the clusters constraints and the geographical constraints at two separate steps, these different aggregation constraints are integrated as a whole for optimal prediction reconciliation [18] [75]. Compared with the state-of-the-art methods, the one-step ahead forecasts of the proposed method for electricity load and solar power data are improved than ordinary multiple stages hierarchy [69] [81].

Big amount of data is generated from energy power consummations meters. Many algorithms are developed with data- driven approach area and target several types of energy applied applications like:

- Load forecasting.
- Prediction.
- Energy pattern profiling.
- Regional energy-consumption mapping.
- Bench-marking for building stocks.
- Global retrofit strategies.
- Guideline making.
- Etc.

Classification methods are developed too in same area of interest like:

- K-mean clustering.
- Self-organizing map.
- Hierarchy clustering.

Both of them (Prediction and Classification) are very important for achieving efficient energy consumption building and enhancing performance. Reducing consumption and environmental impact is also a clear target for applying these strategies. Building prediction and classification models for energy consumption with a very high accurate result requires

very huge efforts. Instead of that, it is useful to build these models by a moderate accuracy. These models have many advantages to be built:

1) Conserve energy based on predicted customer behaviors.
2) Build Demand-Side management (DSM) system according to electrical consummation scenarios.
3) Visualizing predicted and actual energy consumption curves.
4) Implementing a benchmark databases for multilevel consumption.
5) Build a combined model facilitated with designing, running and

http://dx.doi.org/10.21622/ACE.2022.02.1.017

editing functions in the new buildings.

6) offering an image of energy footprint of the building under study.
7) introducing a good imagination for related financial aspects.
8) giving an important tool for decision makers such as:
   - Policymakers.
   - Building owners.
   - Investors.
   - Operators.
   - Engineers.

Dataset of historical energy consumption for a group of customers is used in learning stage to train the algorithm. Pattern of consumption is achieved for each customer to predict next days consumption and to group similar customers in consumption behavior together. Household consumption could be caused from many end-uses devices like heating, ventilation and air-conditioning (HVAC) system, domestic hot water, lighting, plug-loads, elevators, kitchen equipment, ancillary equipment and appliances.

In [11], Load Forecasting is essential in business perspective because, for example, it is important for:

- Power systems planning and operations
- Revenue projection
- Rate design
- Energy trading
- Design evaluation [15]
- Operation strategies [15]
- Enhancing demand and supply management [15]

These fields are always discussed in many sectors of various companies like:

- Electric utilities
- Regulatory commissions
- Industrial and big commercial companies
- Banks
- Trading firms
- Insurance companies

In [15], reducing CO2 emissions, global warming, environmental pollution and energy consumption which are mainly generated from fossil fuel are very hot topics nowadays to save environment and resources. A big percentage of total energy consumption and total CO2 emissions is from buildings. Energy consumption prediction is very important to achieve the international standards for CO2 emissions levels.

D. Field Trends and Work Comparisons

By reviewing the work in the field for the last twenty years, it could be concluded that there are two main types of household customer electrical consumption prediction.

- Resolution: which is the time step of the electrical data.
- Forecast Horizon: which is the window size of the predicted future (day, week, and etc.)

http://dx.doi.org/10.21622/ACE.2022.02.1.017

Resolution and Forecast Horizon, for example could be hourly time step data predicting a horizon of 24 hours in advance. Forecast horizon can be classified as long term (greater than three years), medium (two weeks to three years), and short-term (less than two weeks) shown in [84] and illustrated in Figure 8.

Table 1: shows the Validation Error obtained by the LSTM with DCT, compared other benchmark methods when predicting the test set. It can be seen that the proposed LSTM using the Discrete Cosine Transform DCT significantly improves the validation error obtained by the other prediction models.

Table 1
Validation Error Obtained by The Proposed LSTM Compared with Other Methods [85]

	Validation Error %
LR	7.3395
DT	2.8783
GBT	2.7190
RF	2.2005
DFFN	1.6769
LSTM + CVOA	1.5898
TFT	1.5148
LSTM + Random	1.4472
LSTM + DCT	0.9813

II. CONCLUSION

This paper presented five methods to forecast the energy consumption in a residential building over different time horizons with different time resolutions. Notably, it proposed the use of Deep Learning, more exactly Conditional Restricted Boltzmann Machines and Factored Conditional Restricted Boltzmann Machines for the prediction of energy consumption. The analysis performed showed that FCRBM is a powerful method which outperformed the state-of-the- art prediction methods such as HMM, ANNs, SVMs, RNNs

http://dx.doi.org/10.21622/ACE.2022.02.1.017

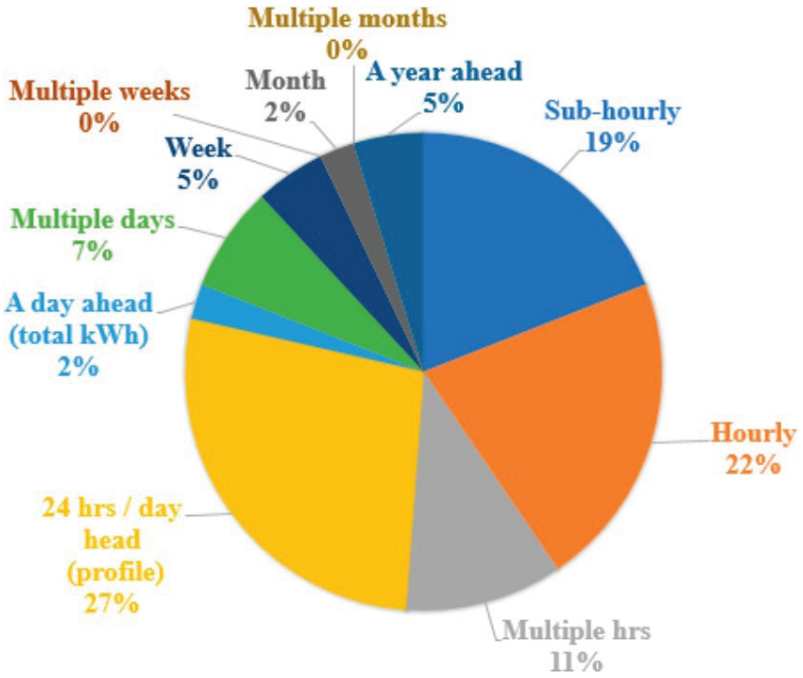


Fig. 8. Compositional breakdown of the forecast horizons [84]

and CRBMs. It is worth mentioning that as the prediction horizon is increasing, FCRBMs and CRBMs seem to be more robust and their prediction error is typically half then that of the ANN. All methods presented showed comparable prediction time, in the order of few hundred milliseconds, and are therefore suitable for near real-time exploitation in applications such as home and building automation systems. From all the experiments, it can be observed that all methods perform better when predicting the aggregated active power consumption than predicting the demand of intermittent appliances(e.g.electric water-heater) recorded with the three sub-metering. Although versatile and successful, CRBMs and FCRBMs come with their own challenges, similar to other ANNs. At this stage, the feasibility of the various methods has been proven. Furthermore, fine-tuning, such as the choice of the optimal number of hidden units or the learn in grate, might improve the performance of these models.

REFERENCES

[1] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah, "A review on time series forecasting techniques for building energy consumption," Renewable and Sustainable Energy Reviews, vol. 74, pp. 902–924, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S1364032117303155

[2] T. Teeraratkul, D. O'Neill, and S. Lall, "Shape-Based Approach to Household Electric Load Curve Clustering and Prediction," IEEE Transactions on Smart Grid, p. 1, 2017.

[3] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering – A decade review," Information Systems, vol. 53, pp. 16–38, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0306437915000733

[4] P. Wang, B. Liu, and T. Hong, "Electric load forecasting with recency effect: A big data approach," International Journal of Forecasting, vol. 32, no. 3, pp. 585–597, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169207015001557

http://dx.doi.org/10.21622/ACE.2022.02.1.017

[5] Y. Goude, R. Nedellec, and N. Kong, "Local Short and Middle Term Electricity Load Forecasting With Semi-Parametric Additive Models," IEEE Transactions on Smart Grid, vol. 5, no. 1, pp. 440–446, 2014.

[6] B.-J. Chen, M.-W. Chang, and C.-J. Lin, "Load forecasting using support vector Machines: a study on EUNITE competition 2001," IEEE Transactions on Power Systems, vol. 19, no. 4, pp. 1821–1830, 2004.

[7] A. Khotanzad, R. Afkhami-Rohani, T.-L. Lu, A. Abaye, M. Davis, and D. J. Maratukulam, "ANNSTLF-a neural-network-based electric load forecasting system," IEEE Transactions on Neural Networks, vol. 8, no. 4, pp. 835–846, 1997.

[8] J. Yang, M. Santamouris, S. E. Lee, and C. Deb, "Energy performance model development and occupancy number identification of institutional buildings," Energy and Buildings, vol. 123, pp. 192 – 204, 2016. [Online]. Available: http:// www.sciencedirect.com/science/ article/pii/S0378778815304552

[9] H. xiang Zhao and F. Magoules, "A review on the prediction of building energy consumption," Renewable and Sustainable Energy Reviews, vol. 16, no. 6, pp. 3586 – 3592, 2012. [Online]. Available: http://www.sciencedirect.com/science/ article/pii/S1364032112001438

[10] Y. Wei, X. Zhang, Y. Shi, L. Xia, S. Pan, J. Wu, M. Han, and X. Zhao, "A review of data-driven approaches for prediction and classification of building energy consumption," Renewable and Sustainable Energy Reviews, vol. 82, pp. 1027–1047, 2018. [Online]. Available: http: //www.sciencedirect.com/science/article/ pii/S136403211731362X

[11] T. Hong and S. Fan, "Probabilistic electric load forecasting: A tutorial review," International Journal of Forecasting, vol. 32, no. 3, pp. 914–938, 2016. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/ S0169207015001508

[12] X. Qiu, P. N. Suganthan, and G. A. J. Amaratunga, "Ensemble incremental learning Random Vector Functional Link network for short-term electric load forecasting," Knowledge-Based Systems, vol. 145, pp. 182–196, 2018. [Online]. Available: http://www.sciencedirect. com/science/article/pii/ S0950705118300236

[13] J.-S. Chou and D.-S. Tran, "Forecasting Energy Consumption Time Series using Machine Learning Techniques based on Usage Patterns of Residential Householders," Energy, 2018. [Online]. Available: http://www.sciencedirect. com/science/article/pii/S0360544218319145

[14] L. Wang, R. Kubichek, and X. Zhou, "Adaptive learning based data-driven models for predicting hourly building energy use," Energy and Buildings, vol. 159, pp. 454–461, 2018. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S0378778817318728

[15] K. Amasyali and N. M. El-Gohary, "A review of data-driven building energy consumption prediction studies," Renewable and Sustainable Energy Reviews, vol. 81, pp. 1192–1205, 2018. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S1364032117306093

[16] P. Lusis, K. R. Khalilpour, L. Andrew, and A. Liebman, "Shortterm residential load forecasting: Impact of calendar effects and forecast granularity," Applied Energy, vol. 205, pp. 654–669, 2017. [Online]. Available: http://www. sciencedirect.com/science/article/ pii/S0306261917309881

[17] J. Luo, T. Hong, and S.-C. Fang, "Benchmarking robustness of load forecasting models under data integrity attacks," International Journal of Forecasting, vol. 34, no. 1, pp. 89–104, 2018. [Online]. Available: http://www.sciencedirect. com/science/article/pii/S0169207017300900

[18] Q. Duan, J. Liu, and D. Zhao, "Short term electric load forecasting using an automated system of model choice," International Journal of Electrical Power & Energy Systems, vol. 91, pp. 92–100, 2017. [Online]. Available: http://www.

http://dx.doi.org/10.21622/ACE.2022.02.1.017

sciencedirect.com/science/article/ pii/S0142061516300710
[19] Z. Wang, Y. Wang, and R. S. Srinivasan, "A novel ensemble learning approach to support building energy use prediction," Energy and Buildings, vol. 159, pp. 109–122, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378778817335272
[20] M. A. M. Daut, M. Y. Hassan, H. Abdullah, H. A. Rahman, M. P. Abdullah, and F. Hussin, "Building electrical energy consumption forecasting analysis using conventional and artificial intelligence methods: A review," Renewable and Sustainable Energy Reviews, vol. 70, pp. 1108–1118, 2017. [Online]. Available: http://www. sciencedirect.com/science/article/pii/S1364032116310619
[21] Z. Wang and R. S. Srinivasan, "A review of artificial intelligence based building energy use prediction: Contrasting the capabilities of single and ensemble prediction models," Renewable and Sustainable Energy Reviews, vol. 75, pp. 796–808, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364032116307420
[22] J. Zhang, Y.-M. Wei, D. Li, Z. Tan, and J. Zhou, "Short term electricity load forecasting using a hybrid model," Energy, vol. 158, pp. 774–781, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S036054421831065X
[23] P. Vrablecová, A. B. Ezzeddine, V. Rozinajová, S. Sárik, and A. K. Sangaiah, "Smart grid load forecasting using online support vector regression," Computers & Electrical Engineering, vol. 65, pp. 102–117, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0045790617320645
[24] Y. Liu, W. Wang, and N. Ghadimi, "Electricity load forecasting by an improved forecast engine for building level consumers," Energy, vol. 139, pp. 18–30, 2017. [Online]. Available: http: //www.sciencedirect.com/science/article/pii/S0360544217313348
[25] Y. T. Chae, R. Horesh, Y. Hwang, and Y. M. Lee, "Artificial neural network model for forecasting sub-hourly electricity usage in commercial buildings," Energy and Buildings, vol. 111, pp. 184–194, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0378778815304102
[26] J. Nowotarski, B. Liu, R. Weron, and T. Hong, "Improving short term load forecast accuracy via combining sister forecasts," Energy, vol. 98, pp. 40–49, 2016. [Online]. Available: http:
//www.sciencedirect.com/science/article/pii/S0360544216000116
[27] G.-F. Fan, L.-L. Peng, and W.-C. Hong, "Short term load forecasting based on phase space reconstruction algorithm and bi-square kernel regression model," Applied Energy, vol. 224, pp. 13–33, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0306261918306238
[28] E. Yukseltan, A. Yucekaya, and A. H. Bilge, "Forecasting electricity demand for Turkey: Modeling periodic variations and demand segregation," Applied Energy, vol. 193, pp. 287–296, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306261917301848
[29] G.-F. Fan, L.-L. Peng, W.-C. Hong, and F. Sun, "Electric load forecasting by the SVR model with differential empirical mode decomposition and auto regression," Neurocomputing, vol. 173, pp. 958–970, 2016. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S0925231215012230
[30] Y. Chen, H. Tan, and X. Song, "Day-ahead Forecasting of Non- stationary Electric Power Demand in Commercial Buildings: Hybrid Support Vector Regression Based," Energy Procedia, vol. 105, pp. 2101–2106, 2017. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S1876610217306422
[31] X. Zhang, J. Wang, and K. Zhang, "Short-term electric load forecasting based on singular spectrum analysis and support vector machine optimized by Cuckoo search algorithm," Electric Power Systems Research, vol. 146, pp. 270–285, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/

S0378779617300445
[32] S. Fan and R. J. Hyndman, "Short-Term Load Forecasting Based on a Semi-Parametric Additive Model," IEEE Transactions on Power Systems, vol. 27, no. 1, pp. 134–141, 2012.
[33] N. Ghadimi, A. Akbarimajd, H. Shayeghi, and O. Abedinia, "Two stage forecast engine with feature selection technique and improved meta-heuristic algorithm for electricity load forecasting," Energy, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0360544218313859
[34] A. I. Saleh, A. H. Rabie, and K. M. Abo-Al-Ez, "A data mining based load forecasting strategy for smart electrical grids," Advanced Engineering Informatics, vol. 30, no. 3, pp. 422–448, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S1474034616301331
[35] F. Zhang, C. Deb, S. E. Lee, J. Yang, and K. W. Shah, "Time series forecasting for building energy consumption using weighted Support Vector Regression with differential evolution optimization technique," Energy and Buildings, vol. 126, pp. 94–103, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378778816303899
[36] A. Lahouar and J. B. H. Slama, "Day-ahead load forecast using random forest and expert input selection," Energy Conversion and Management, vol. 103, pp. 1040–1051, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0196890415006925
[37] Y. Chen, P. Xu, Y. Chu, W. Li, Y. Wu, L. Ni, Y. Bao, and K. Wang, "Short-term electrical load forecasting using the Support Vector Regression (SVR) model to calculate the demand response baseline for office buildings," Applied Energy, vol. 195, pp. 659–670, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0306261917302581
[38] S. Mishra and V. K. Singh, "Monthly Energy Consumption Forecasting Based On Windowed Momentum Neural Network," IFAC- PapersOnLine, vol. 48, no. 30, pp. 433–438, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2405896315030591
[39] K. Gajowniczek and T. Zabkowski, "Short Term Electricity Forecasting Using Individual Smart Meter Data," Procedia Computer Science, vol. 35, pp. 589–597, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050914011053
[40] M. Mordjaoui, S. Haddad, A. Medoued, and A. Laouafi, "Electric load forecasting by using dynamic neural network," International Journal of Hydrogen Energy, vol. 42, no. 28, pp. 17 655–17 663, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0360319917310455
[41] W.-C. Hong, "Electric load forecasting by support vector model," Applied Mathematical Modelling, vol. 33, no. 5, pp. 2444–2454, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0307904X08001844
[42] M. Rana and I. Koprinska, "Forecasting electricity load with advanced wavelet neural networks," Neurocomputing, vol. 182, pp. 118–132, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0925231215019141
[43] J.-S. Chou and N.-T. Ngo, "Time series analytics using sliding window metaheuristic optimization-based machine learning system for identifying building energy consumption patterns," Applied Energy, vol. 177, pp. 751–770, 2016. [Online]. Available: http: //www.sciencedirect.com/science/article/pii/S0306261916306717
[44] C. Kuster, Y. Rezgui, and M. Mourshed, "Electrical load forecasting models: A critical systematic review," Sustainable Cities and Society, vol. 35, pp. 257–270, 2017. [Online]. Available: http: //www.sciencedirect.com/science/article/pii/S2210670717305899
[45] S. Aman, Y. Simmhan, and V. K. Prasanna, "Holistic Measures for Evaluating Prediction Models in Smart Grids," IEEE Transactions on Knowledge and Data

Engineering, vol. 27, no. 2, pp. 475–488, 2015.
[46] J.-S. Chou and N.-T. Ngo, "Smart grid data analytics framework for increasing energy savings in residential buildings," Automation in Construction, vol. 72, pp. 247–257, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0926580516000030
[47] M. Q. Raza and A. Khosravi, "A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings," Renewable and Sustainable Energy Reviews, vol. 50, pp. 1352–1372, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S1364032115003354
[48] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, "Unsupervised real-time anomaly detection for streaming data," Neurocomputing, vol. 262, pp. 134–147, 2017. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S0925231217309864
[49] C. Lynch, M. J.O'Mahony, and R. A. Guinee, "Electrical Load Forecasting Using An Expanded Kalman Filter Bank Methodology," IFAC- PapersOnLine, vol. 49, no. 25, pp. 358–365, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2405896316327070
[50] Z. Guo, K. Zhou, X. Zhang, and S. Yang, "A deep learning model for short-term power load and probability density forecasting," Energy, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0360544218313872
[51] W. He, "Load Forecasting via Deep Neural Networks," Procedia Computer Science, vol. 122, pp. 308–314, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050917326170
[52] N. Ding, Y. Besanger, and F. Wurtz, "Next-day MV/LV substation load forecaster using time series method," Electric Power Systems Research, vol. 119, pp. 345–354, 2015. [Online]. Available: http: //www.sciencedirect.com/science/article/pii/S0378779614003575
[53] A. S. Ahmad, M. Y. Hassan, M. P. Abdullah, H. A. Rahman, F. Hussin, H. Abdullah, and R. Saidur, "A review on applications of ANN and SVM for building electrical energy consumption forecasting," Renewable and Sustainable Energy Reviews, vol. 33, pp. 102–109, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S1364032114000914
[54] H. Jiang, Y. Zhang, E. Muljadi, J. J. Zhang, and D. W. Gao, "A Short-Term and High-Resolution Distribution System Load Forecasting Approach Using Support Vector Regression With Hybrid Parameters Optimization," IEEE Transactions on Smart Grid, vol. 9, no. 4, pp. 3341–3350, 2018.
[55] X. Fu, X.-J. Zeng, P. Feng, and X. Cai, "Clustering-based short-term load forecasting for residential electricity under the increasing-block pricing tariffs in China," Energy, vol. 165, pp. 76–89, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0360544218319285
[56] W. Ma, S. Fang, G. Liu, and R. Zhou, "Modeling of district load forecasting for distributed energy system," Applied Energy, vol. 204, pp. 181–205, 2017. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S0306261917308747
[57] H. Shiraki, S. Nakamura, S. Ashina, and K. Honjo, "Estimating the hourly electricity profile of Japanese households – Coupling of engineering and statistical methods," Energy, vol. 114, pp. 478–491, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0360544216311227
[58] L. M. Candanedo, V. Feldheim, and D. Deramaix, "Data driven prediction models of energy use of appliances in a low-energy house," Energy and Buildings, vol. 140, pp. 81–97, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378778816308970
[59] S. Naji, A. Keivani, S. Shamshirband, U. J. Alengaram, M. Z. Jumaat, Z. Mansor, and M. Lee, "Estimating building energy consumption using extreme learning machine method," Energy, vol. 97, pp. 506–516, 2016. [Online]. Available: http://

http://dx.doi.org/10.21622/ACE.2022.02.1.017

www.sciencedirect.com/science/article/ pii/S036054421501587X
[60] C. Tu, X. He, Z. Shuai, and F. Jiang, "Big data issues in smart grid – A review," Renewable and Sustainable Energy Reviews, vol. 79, pp. 1099–1107, 2017. [Online]. Available: http: //www.sciencedirect.com/science/article/pii/S1364032117307748
[61] O. C. Ozerdem, E. O. Olaniyi, and O. K. Oyedotun, "Short term load forecasting using particle swarm optimization neural network," Procedia Computer Science, vol. 120, pp. 382–393, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050917324663
[62] A. R. Khan, A. Mahmood, A. Safdar, Z. A. Khan, and N. A. Khan, "Load forecasting, dynamic pricing and DSM in smart grid: A review," Renewable and Sustainable Energy Reviews, vol. 54, pp. 1311–1322, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S136403211501196X
[63] C. Tong, J. Li, C. Lang, F. Kong, J. Niu, and J. J. P. C. Rodrigues, "An efficient deep model for day-ahead electricity load forecasting with stacked denoising auto-encoders," Journal of Parallel and Distributed Computing, vol. 117, pp. 267–273, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S074373151730196X
[64] Y. Yang, J. Che, Y. Li, Y. Zhao, and S. Zhu, "An incremental electric load forecasting model based on support vector regression," Energy, vol. 113, pp. 796–808, 2016. [Online]. Available: http: //www.sciencedirect.com/science/article/pii/S0360544216310167
[65] Y. Chen, Y. Yang, C. Liu, C. Li, and L. Li, "A hybrid application algorithm based on the support vector machine and artificial intelligence: An example of electric load forecasting," Applied Mathematical Modelling, vol. 39, no. 9, pp. 2617–2632, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0307904X14005769
[66] C. Robinson, B. Dilkina, J. Hubbs, W. Zhang, S. Guhathakurta, M. A. Brown, and R. M. Pendyala, "Machine learning approaches for estimating commercial building energy consumption," Applied Energy, vol. 208, pp. 889–904, 2017. [Online]. Available: http: //www.sciencedirect.com/science/article/pii/S0306261917313429
[67] L. Hernández, C. Baladron, J. M. Aguiar, B. Carro, A. Sánchez- Esguevillas, and J. Lloret, "Artificial neural networks for short-term load forecasting in microgrids environment," Energy, vol. 75, pp. 252– 264, 2014. [Online]. Available: http://www.sciencedirect.com/science/ article/pii/S0360544214008871
[68] I. M. Coelho, V. N. Coelho, E. J. da S. Luz, L. S. Ochi, F. G. Guimarães, and E. Rios, "A GPU deep learning metaheuristic based model for time series forecasting," Applied Energy, vol. 201, pp. 412– 418, 2017. [Online]. Available: http://www.sciencedirect.com/science/ article/pii/S0306261917300041
[69] T. Yang, M. Ren, and K. Zhou, "Identifying household electricity consumption patterns: A case study of Kunshan, China," Renewable and Sustainable Energy Reviews, vol. 91, pp. 861–868, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/ S136403211830265X
[70] A. Boustani, A. Maiti, S. Y. Jazi, M. Jadliwala, and V. Namboodiri, "Seer Grid: Privacy and Utility Implications of Two-Level Load Prediction in Smart Grids," IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 2, pp. 546–557, 2017.
[71] J. Yang, C. Ning, C. Deb, F. Zhang, D. Cheong, S. E. Lee, C. Sekhar, and K. W. Tham, "k-Shape clustering algorithm for building energy usage patterns analysis and forecasting model accuracy improvement," Energy and Buildings, vol. 146, pp. 27–37, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378778817305352
[72] M. W. Ahmad, M. Mourshed, and Y. Rezgui, "Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building

energy consumption," Energy and Buildings, vol. 147, pp. 77–89, 2017. [Online]. Available: http://www.sciencedirect. com/science/article/pii/S0378778816313937

[73] S. Deng, C. Yuan, L. Yang, and L. Zhang, "Distributed electricity load forecasting model mining based on hybrid gene expression programming and cloud computing," Pattern Recognition Letters, vol. 109, pp. 72–80, 2018. [Online]. Available: http://www.sciencedirect. com/science/article/pii/S0167865517303665

[74] Y. Chen, M. Kloft, Y. Yang, C. Li, and L. Li, "Mixed kernel based extreme learning machine for electric load forecasting," Neurocomputing, vol. 312, pp. 90–106, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231218306532

[75] W. Yu, F. Zhuang, Q. He, and Z. Shi, "Learning deep representations via extreme learning machines," Neurocomputing, vol. 149, pp. 308–315, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0925231214011461

[76] Z. Ding and M. Fei, "An Anomaly Detection Approach Based on Isolation Forest Algorithm for Streaming Data using Sliding Window," IFAC Proceedings Volumes, vol. 46, no. 20, pp. 12–17, 2013. [Online]. Available: http://www.sciencedirect. com/science/article/ pii/S1474667016314999

[77] C. Fan, F. Xiao, Y. Zhao, and J. Wang, "Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data," Applied Energy, vol. 211, pp. 1123–1135, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/ pii/S0306261917317166

[78] W. Zhang, H. Quan, and D. Srinivasan, "Parallel and reliable probabilistic load forecasting via quantile regression forest and quantile determination," Energy, 2018. [Online]. Available: http: //www.sciencedirect.com/science/article/pii/S0360544218313148

[79] M. Shepero, D. van der Meer, J. Munkhammar, and J. Widen, "Residential probabilistic load forecasting: A method using Gaussian process designed for electric load data," Applied Energy, vol. 218, pp. 159–172, 2018. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S030626191830299X

[80] C. Lopez, W. Zhong, and M. Zheng, "Short-term Electric Load Forecasting Based on Wavelet Neural Network, Particle Swarm Optimization and Ensemble Empirical Mode Decomposition," Energy Procedia, vol. 105, pp. 3677–3682, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1876610217309323

[81] G. Chicco, R. Napoli, and F. Piglione, "Comparisons among clustering techniques for electricity customer classification," IEEE Transactions on Power Systems, vol. 21, no. 2, pp. 933–940, 2006.

[82] G. Dudek, "Neural networks for pattern-based short-term load forecasting: A comparative study," Neurocomputing, vol. 205, pp. 64– 74, 2016. [Online]. Available: http://www.sciencedirect.com/science/ article/pii/S0925231216302764

[83] V. Thouvenot, A. Pichavant, Y. Goude, A. Antoniadis, and J. M. Poggi, "Electricity Forecasting Using Multi-Stage Estimators of Nonlinear Additive Models," IEEE Transactions on Power Systems, vol. 31, no. 5, pp. 3665–3673, 2016.

[84] J. Runge and R. Zmeureanu, "A review of deep learning techniques for forecasting energy use in buildings," Energies 2021, vol. 14, p. 608, 2021. [Online]. Available: https://www.mdpi.com/1996-1073/14/3/608/ pdf

[85] F. M.-A. J. F. Torres and A. Troncoso, "A deep lstm network for the spanish electricity consumption forecasting," Neural Computing and Applications, 2021. [Online]. Available: https://link.springer.com/ content/pdf/10.1007/s00521-021-06773-2.pdf

http://dx.doi.org/10.21622/ace.2022.02.1.043

Algorithms for Comparing Large Pedigree Graphs

Nahla A. Belal[1], Lamiaa A. Amar[2] and Hany H. Sherief[2]

[1]College of Computing and Information Technology, Arab Academy for Science, Technology and Maritime Transport (AASTMT), Smart Village, B 2401, Giza, Egypt

[2]Alexandria University, Faculty of Science, Al Azaritah WA Ash Shatebi, Qism Bab Sharqi, Alexandria, Egypt

Email: { nahlabelal@aast.edu, lamy2003@hotmail.com, hanysherief@gmail.com }

Received on, 08 May 2022 - Accepted on, 13 June 2022 - Published on, 29 June 2022

ABSTRACT

The importance of pedigrees is translated by geneticists as a tool for diagnosing genetic diseases. Errors resulting during collection of data and missing information of individuals are considered obstacles in deducing pedigrees, especially larger ones. Therefore, the reconstructed pedigree graph evaluation needs to be undertaken for relevant diagnosis. This requires a comparison between the derived and the original data. The present study discusses the isomorphism of huge pedigrees with labeled and unlabeled leaves, where a pedigree has hundreds of families, which are monogamous and generational. The algorithms presented in this paper are based on a set of bipartite graphs covering the pedigree and the problem addressed is parameter tractable. The Bipartite graphs Covering the Pedigree (BCP) problem is said to possess a time complexity of $f(k).mod(X)^{O(1)}$ where f is the computing function that grows exponentially. The study presents an algorithm for the BCP problem that can be categorized as a polynomial-time-tractable evaluation of the reconstructed pedigree. The paper considers pedigree graphs that consist of both labeled and unlabeled leaves that make use of parameterized and kernelization algorithms to solve the problem. The kernelization algorithm executes in $O(k^3)$ for the BCP graphs.

Keywords: Pedigree Graphs, Isomorphism, Parameterized Algorithm, Kernelization Algorithms, Bipartite graphs Covering Peigrees (BCP) Problem

I. INTRODUCTION

Recently, an enormous amount of populace genotype data produced from a Single-Nucleotide Polymorphism (SNP) in genomes were collected. The amount of such data are expected to rise rapidly in the near future. The populace genotype data relationships are normally represented using a family tree or pedigree. A pedigree depicts the relationship amongst the family members and portrays a specific trait, abnormality, or a disease commonly shared by them through inheritance [1]. A pedigree is always of great interest and importance to scientists. For example, pedigree analysis can be be used for initiating a country-wise exchange of germplasm [2]. Further, it can also serve to devise strategies accordingly. Approaches to reconstruct pedigree are multi-fold. One such way is to construct using genetic data of survivors and backtracking to identify the ancestral origins [3, 4, 5, 6]. Reconstruction methods incorporate comparing the derived pedigree against the original, as both possess the same set of individuals and genetic data but are mutually exclusive in their inferred ancestors. The existing methods used to compare pedigrees are very low on accuracy [7, 8, 9].

http://dx.doi.org/10.21622/ace.2022.02.1.043

A team of researchers have developed a method for comparing the topology of pedigrees with labeled individuals [10], where they studied the following two main pedigree graph comparison aspects:

- Evaluation of the reconstructed pedigree in comparison with the original, and
- Examination of the inferred and true ancestors for isomorphism or exhibition of dissimilitude

The authors employed the edit distance algorithm. Both problems are said to be computationally complex and NP-hard. The edit-distance problem is APX-hard, meaning that there is a Polynomial Time Approximation Scheme (PTAS) reduction from every problem in APX to the former and is highly specific in the case of graph matching.

Chen, in his research presented models for a pedigree with labeled leaves [11]. Labeled leaves represent individuals that have available DNA on which genotyping or sequencing can be performed without relying on genealogical structure. He also compared two pedigrees with labeled leaves only. Avoiding pedigrees with leaves missing the data of the biological father, that are unlabeled leaves, in turn, has considerably lowered the accuracy in the results [12]. Jiang, on the other hand, presented his research work to evaluate pedigrees with unlabeled leaves and proved it to be GI-hard and Fixed-Parameter Tractable (FPT) [13]. The large running time of this method is the major drawback hampering its comparison to arbitrary 2-generation pedigrees. Similarly, Amar et al. studied the isomorphism for large unlabeled sub-pedigree, but only up to 7-generation, by using two traits and proved that the fixed parameters are tractable [14]. Recently, [15] developed the PedigreeNet viewer which was able to generate a pedigree network of 4706 maize lines and 5487 relationships.

A lot of practical work recently carried out show the massive importance of pedigrees. Among the most recent work done is Xie et al. [16], where the studied a chinese pedigree for retinal vasculopathy and how cerebral leukoencephalopathy can be deduced. Also, in [17] developed an R package that uses pedigree and genomic data to analyze genetic connectedness. Pedigree analysis also extends to plants, in [18], the authors studied 7 generation pedigrees on peaches.

This research focuses on the algorithm for large pedigree isomorphism problem that requires the connected components in a given Family F to constitute a graph. Vertices of the graph can be separated into two sets that are dissimilar and independent, with edges connecting these two disjoint sets of vertices. The disjoint sets of vertices and edges interconnecting them contain labeled and unlabeled leaves, and are represented as an isomorphism problem, between the reconstructed and the original pedigree. In this paper, an algorithm is proposed, for the bipartite graphs constituting the pedigree for which leaves are labeled and unlabeled. The algorithm is executed in polynomial time, given that the bipartite graphs in the family are bound by a constant. Along with the above, a kernelization problem whose time complexity increases linearly is presented that returns an instance of size $O(k^3)$.

http://dx.doi.org/10.21622/ace.2022.02.1.043

The following are the definitions based on which this research has been conducted:

Definition 1 A pedigree $G = (P, X, g, l)$ consists of a directed and acyclic graph where P denotes set of vertices and edges represented as: $P = (I(P);E(P))$;
the gender function denoted as g and represented as: $g : I(P) \rightarrow \{male, female\}$
X the set of labelled vertices and that which is denoted as $X \subseteq I(P)$ and
l denoted for injective labelling as $l : X \rightarrow N$
where N is the graph size and the set of vertices in P have an in-degree of either 0 or 2 and if $(a, v1)$ and $(b, v1) \in$ E
then g(a) $\neq$ g(b).

The set of vertices and directed edges represent the parent-child relationship in this graph and the graph is acyclic in nature. A monogamous vertex "v" in graph P and pedigree G is defined to be that vertex which mates with the opposite gender vertex exactly once and is formally defined as a $v \neq v^{'}$ where (v, x) and $(v^{'}, x) \in$ to $E(P)$ for some x in the set of vertices $I(P)$ is exactly one. If all the vertices in that graph mate with exactly one other vertex of opposite gender then the pedigree is said to be monogamous.

A pedigree $G = (P, X, g, l)$ is generational if there exists $Q : I(P) \rightarrow N$ and $Q(v) = 1$, where v has an in-degree of 0, and there exists an edge (u, v) in $E(P)$, such that the in-degree of v is more than 0 and $Q(v) = Q(u) + 1$. Then Q indicates the generation for v which is nothing but a numeral.

Definition 2 Two pedigree graphs $G = (P, X, g, l)$, G0 $= (P^{'}, X^{'}, g^{'}, l^{'})$, with labelled and unlabelled leaves are said to be isomorphic when the following condition holds true. There exists a gender function $g : I(P) \rightarrow (male, female)$, for the labelled and unlabelled leaves, the gender is unknown and there exists a bijection, i.e. one to one and onto mapping between the two sets. It is represented as : $\psi : V \rightarrow V^{'}$, if for every $v \in V$ the gender of v i.e. $g(v)$ matches the gender of v in the other set, i.e $g'(v)$ and $(x, v) \in E$ if, ψ(x) ψ(v) $\in E^{'}$.

Definition 3 Maximum : 3-Dimensional Matching : Given a labelled pedigree graph P constituting of sets of disjoint paths of size n, termed A, B and C, and a set of triples $Y \subseteq A \times B \times C$. M is defined as the subset of Y such that every 2 triples in M are disjoint and cardinality of $mod[M]$ is maximized. Given an instance of two pedigree P and $P^{'}$. An arbitrary fixed order of elements is taken into consideration i.e. $Y = A \cup B \cup C$ and use the assigned order. For each triplet $t = (xi, yj, zk)$ a sub pedigree containing (x_i, y_j, z_k) of the pedigree P as follows:

- A female individual is created with (x_i, y_j, z_k) for a leaf of P.
- Create parents w_h^t, m^t for (x, y, z) and h is the generation of the pedigree for $2 \leq$ mod $(A \cup B \cup C)$
- For each m_1^t, m_2^t and w_2^t are the parents who are created upto generation h and for each w_1^t create parents u_1^t and v_1^t upto generation h. P now contains Y sub pedigrees and each has a unique leaf.
- Next $P^{'}$ is initialized as a copy of P and the stated condition is $1 \leq h \leq$ mod $(A \cup B \cup C)$.
- t_1, t_2, t_3 sharing h items in mod $(A \cup B \cup C)$ and u_1^t.....u_n^t are merged and v_1^t.....v_n^t are merged to form sub-pedigree with max isomorphism.

http://dx.doi.org/10.21622/ace.2022.02.1.043

Problem Definition Suppose there exists a derived pedigree graph. Hence, a framework to compare large real and induced pedigrees will be discussed. A sample pedigree is shown in Figure 1. To compare the two pedigrees, the sub pedigrees of this graph have to be taken into consideration. The sub pedigrees are represented in Figure 2. Where, Figure 2(a) and 2(c) constitute labelled leaves, whereas 2(b), 2(d) and 2(e) show unlabelled leaves. These sub pedigrees exude generational and monogamous trait. Considering the pedigree graph shown in Figure 1, it will be subsequently proved in 3 that that there exists isomorphism upto the 3rd generation. If there exists an isomorphism between the deduced and the original pedigree with both labeled and unlabeled leaves, it will be validated. This can help geneticist to use it as an essential tool to diagnose the genetic disease. Thus, the problem statement is as follows: The reconstructed or obtained graph of a large pedigree can be validated by evaluating the existence of isomorphic traits between its sub-pedigrees that are bound by the linearly increasing time complexity of $O(k^3)$.

The problem described can be formally defined as follows.

INSTANCE

A family $F_0 = (V_{F0}, E_{F0})$ consisting of k_0 disjoint sub-families, and a real pedigree graph $P = (V_p, E_p)$, with the same number of individuals $jV_{F0}j = jV_Pj$, and a large sub-family $K_{p,d}$.

SOLUTION

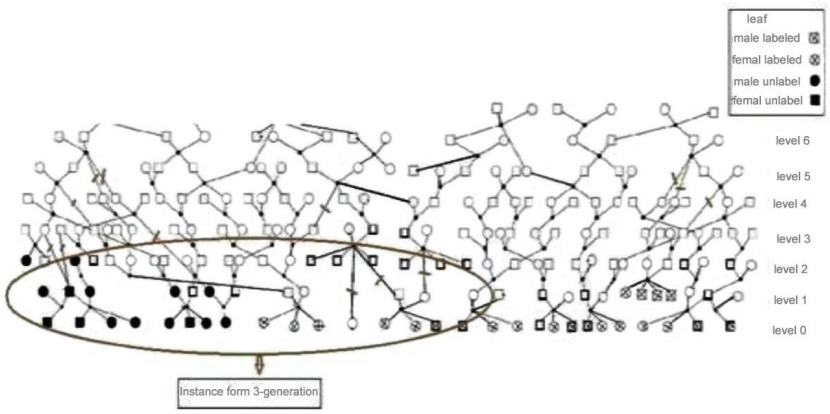


Figure 1: A large pedigree containing labeled and unlabled leaves.

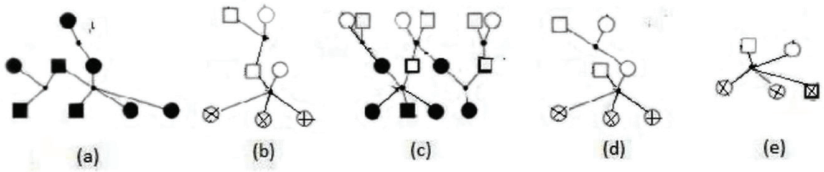


Figure 2: (a) and (c) constitute of labelled leaves whereas (b), (d) and (e) have unlabelled leaves.

F_0 is a sub-pedigree of G up to three-generations. This study discusses the isomorphism of huge pedigrees with labelled and unlabeled leaves, where the pedigree has hundreds of families, that are monogamous and generational.

http://dx.doi.org/10.21622/ace.2022.02.1.043

The main contributions of the paper are as summarized in the points below:

- The problem of comparing large pedigree graph is defined.
- Algorithms for solving the problem are designed.
- Faster kernelization algorithms for solving the problem are designed.
- Theorems and lemmas for proving and analyzing the proposed algorithms are given.
- The importance of pedigrees to geneticists is highlighted.

This paper is organized in a way that succeeding sections present the problem and the results followed by implications and limitations of the study. Section 2 summarizes the methods used in this paper. The algorithms are presented in Section 3 along with their analysis. Finally, Section 4 concludes and sets future directions.

2. METHODS

Kernelization algorithms aim at data reduction, by reducing the input to a function of the parameter. Hence, they will be used to remove the sub-families from the reconstructed pedigree till end and the similarity between the sub-pedigrees of the original and the reconstructed pedigree will be established. The algorithms execute in polynomial time across large pedigree graphs and the total number of individuals in sub-pedigrees would be bounded by $O(k^3)$.

The researchers have considered a set of BCP graphs. The parameterized algorithms are used to solve the BCP problem and the kernelization algorithm is applied repeatedly, whenever there is a sub-family that has to be removed. At the end of the process, no sub-families can be removed. At this stage the instances of each pedigree are evaluated to identify the isomorphism. This methodology can help in diagnosing the genetic diseases across the generations in large pedigrees containing both labeled and unlabeled leaves. Using bipartite graphs for which the CUPB algorithm runs in polynomial time; instead of the Fixed Parameter Tractable algorithm, significantly reduces the time required to compare pedigrees till the second generation. However, this is not possible owing to large amounts of missing datasets leading to difficulty in reconstruction. Also, due to the low-level accuracy while reconstructing the pedigree and detecting the isomorphic trait.

Section 3 presents the algorithm, a set of Bipartite graphs Covering the Pedigree (BCP), and it is shown to be fixed-parameter tractable that can be solved in $|n|0(1)$. Moreover, parameterized algorithms and kernelization algorithms for the problem defined are also presented.

3. RESULTS

This section presents the algorithms developed to solve the defined problem, along with the analysis of each algorithm.

http://dx.doi.org/10.21622/ace.2022.02.1.043

3.1. A Set of Bipartite Graphs Covering the Pedigree Graphs (BCP)

Before formulating the problem solution, it is required to introduce some definitions before using the concept. Let (F_0, P) be an illustration of a BCP problem, where the total number of individuals in the family F_0 is equal to the number of individuals in pedigree P. It is required to prove that the total number of subfamilies in F_0 forms a sub-pedigree of P. As per the research conducted in [14], F_0 becomes a sub pedigree of P if and only if removal of K - 1 edges from pedigree P transforms it into F_0. The approach taken in this paper is the creation of sub families, succeeded by pairing of sub families using PairBipartite, which ensures that if the family F^* comprises of a sub-family that is constituted of a connected component, then it is possible to identify a sub-family in F_0 and conclude its image in F^*. The proof to the above is illustrated below:

Input Two families F_0 and F^* have the same number of individuals, where F_0 is a collection of disjoint sub families.

Parameter K, where $K = K_0$ - F^* and K_0 and K^* are the number of connected components in F_0 and F^*. The number of edges in F^* that are not in F_0 would be equal to the number of edges which when removed would transform F^* to F_0 family.

Lemma 1 *Let F_i be a family with K tuple leaves $(l_1, l_2, ...l_k)$ in the same family. Let the labelled leaves be denoted from $1 \leq i \leq s$ and the unlabelled leaves be $s + 1 \leq i \leq K$. If u and v are two individuals and $u \neq v$, and u and v are connected by edges $(l_{u1}, ...l_{us})$, $(l_{vs+1}, ...l_{vk})$, and are not equal, then u and v are parents of the children $(l_{ui})_{i=1\ to\ s}$ and $(l_{vi})_{i=s+1\ to\ k}$.*

The sub family algorithm then takes into account the leaves and their corresponding information of parents and labels and separates them into sub families $k_{m,n}$.

3.1.1. Sub-Family Algorithm $(F, k_{m,n})$

Input: Instance of sub-pedigree F, where g is the gender function and $g =$ {male, female} or $g =$ null. X belongs to the set of vertices $(I(P))$, which are leaves, w_i are the labelled leaves, $1 \leq i \leq$K, and t is the set of unlabelled leaves. A set of sub-families $k_{m,n}$ is created as follows :

1. Let u be an individual in a sub-pedigree F
2. For all the individuals in this sub-pedigree F
3. If u has out degree $= \{w_i\}$ and in-degree $= \emptyset$ then

   {

   If $w_i \in X$ //(marks the presence of labelled leaves) then

   Store $m \leftarrow u$ and $n \leftarrow l_{wi}$;

   Elseif $w_{i-t} \in X$ and $w_t \in X$ // (marks the presence of unlabelled and labelled leaves) then

   Store $m \leftarrow u$ and $n \leftarrow l_{wi}$ and n_t null

   Else Store $m \leftarrow u$ and $n \leftarrow g(null)$; // marks the presence of unlabelled leaves only

   }

http://dx.doi.org/10.21622/ace.2022.02.1.043

4. Else Delete $m+n$ from F // here u is a leaf and its parent is v
5. If $F \neq \ominus$ then return to step 2;
6. Else Stop

3.1.2. Pairing Sub-Families in F^0 and F^*

F^* has a connected set of sub families as a bipartite graph. If F_0 is a sub-pedigree of F^*, then a sub family of F_0 can be directly mapped to a sub family of F^*. This is done by Pair Bipartite algorithm. An example is shown in Figure 3.

Select the largest sub-family $K_{p,d}$ that belongs to F^* and map with a sub family from F_0. In case of labelled leaves, leaves with similar labels are retained while those that are unlabelled have edges removed to transform the sub family of F_0 into a sub family of F^*. $K_{p,d}$ is transformed into $K_{v,w}$ as shown in the example Figure 4.

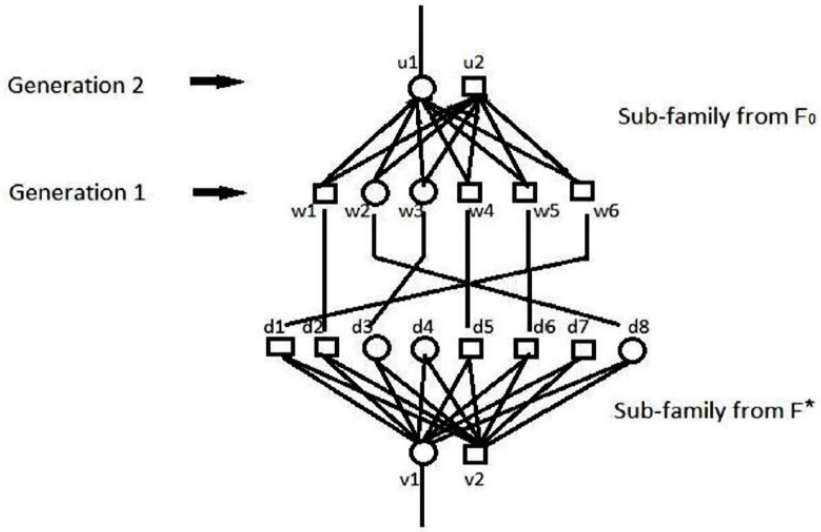


Figure 3: In the case of a labeled leaves: Determine the similar leaves between two sub-families from F_0 and F^* and remove the leaves with different labels.

PairBipartite(F_0,F^*,$K_{p,d}$)
Input : F_0, F^* are sub families and $K_{p,d}$ is the sub family
// t = the number of labelled leaves in the sub-family in F^*
// b = the set of sub families in F_0
b = Sub Family Algorithm(F_0, $K_{v,w}$)
Suppose b is the set of sub families in F_0 and the sub families are smaller than $K_{p,d}$;
If b = null then F_0 is not a sub pedigree of F^*
Else the largest sub family $K_{v,w}$ in F_0 is selected
{ If there exists labelled leaves in the sub family then
Remove $d-w$ leaves (i.e. leaves that are in d and not in w) and have different labels such that
$K_{p,w}$ becomes a sub family of F^*
Else if there exists labelled and unlabeled leaves in a sub family present in set b then

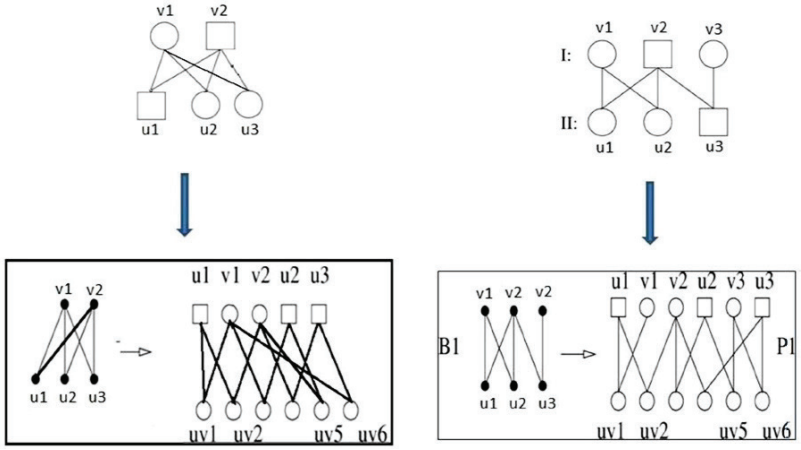


Figure 4: Mapping two sub-families with unlabeled leaves by using the MCIP algorithm in Amar et al. [14]

Remove $d-w$ leaves in $K_{p,d}$ those whose names are null or have different names
Else if unlabelled leaves are present in the sub family then
Remove $d-w$ edges;
Convert $K_{v,w}$ in F_0 and $K_{p,d}$ in F^* that forms the sub families of F_0' and F^*
Initialize empty set $E(f_1)$, $I(f_1)$
Suppose v is marked as the male vertex where $v \geq 2$;
Suppose w is marked as the female vertex where $w \geq 0$;
F_1 the new sub-family is designated as $(I(f_1), E(f_1))$
Generation 1: $I(f_1) \leftarrow v \cup w$
Generation 2: replace each edge between v and w by a new leaf vertex which is marked as female
$\mathrm{E}(F_1) \leftarrow E(f_1) \cup \{(v, vw), (w, vw)\}$
Repeat the steps to convert the sub tree $K_{p,w}$ in F^* into f_2 denoted by $(I(f_2), E(f_2))$, Match f_1 and f_2 using the MCIP algorithm as discussed by Jiang [13] to exhibit isomorphism across pedigrees and levels. Justifying that evaluation of the reconstructed pedigree exuding the isomorphic behaviour to the original can be used by geneticists accordingly.

The validity of the Algorithm Bipartite is embedded in Lemma 2, which presents an assumption and substantiates it with a proof:

Lemma 2 *(F_0,F^*,$K_{p,d}$) outputs two sub-families $F_{0'}$ and $F^{*'}$ which are sub pedigrees of F_0 and F^*, respectively. Then F_0 is a sub pedigree of F^* if and only if $F_{0'}$ is a sub pedigree of $F^{*'}$.*

Proof. Initially the sub family algorithm (F_0, $K_{m,n}$) is executed to determine all the sub families in F_0 and store the information of the sub families.

"If b = null, then F_0 is not a sub pedigree of F^*" can be proved through contradiction. Consider that σ denotes isomorphic mapping from F_0 to a sub pedigree of F^*. Then σ would chart a vertex or an individual in F_0 to the parent of sub family $K_{v,w}$ in F^*. The possible cases that v can be either parent or leaf of the sub family $K_{p,d}$ proves that F_0 has a sub family not larger than $K_{p,d}$. This contradicts the proposition that b is null. Let there be two sub families that are denoted as outputs $F_{0'}$ and $F^{*'}$. Execution of the Step 4 of the Pair Bipartite Algorithm removes some edges from F^* and then removes two identical families, one from F_0 and F^*. So $F_{0'}$ and $F^{*'}$ can be written as $F_{0'} = F_0/K_{v.,w}$ and $F^{*'} = F^*/K_{v,w}$ and in the case of $F^{*'}$ all the edges incident to

http://dx.doi.org/10.21622/ace.2022.02.1.043

d in $K_{p,d}$ but not to w in $K_{v,w}$ are removed. This proves that $F_{0'}$ and $F^{*'}$ are sub pedigree of F_0 and F^*. Since the $F_{0'}$ and $F^{*'}$ are the sub pedigrees of F_0 and F^*, if an identical sub family is removed then if F_0 is not a sub pedigree of F^*, then $F_{0'}$ cannot be a sub pedigree of $F^{*'}$.

Now, if F_0 is a sub-pedigree of F^* and σ is an isomorphic mapping from F_0 to a sub-pedigree of F^*, the mapping σ must map a sub-family in F_0 to the sub-family in F^*. Select the largest sub family $K_{v,w}$ from the sub pedigree F_0 and perform the following operations :

- In case of the labelled leaves, the number of edges that have to be removed is $d-w$ edges, where edges starting from d and connecting w leaves should be of the same gender, otherwise they have to be removed.
- In case of unlabeled leaves, remove $d-w$ edges
- In case of labelled and unlabelled leaves in the sub family, save the edges with common gender and remove the edges which traverse to unlabelled leaves.

If the two sub families are similar, then it is validated that a sub family in F_0 can be mapped onto sub family of F^*. To validate isomorphism, it is required to prove that $K_{v,w1}$ and $K_{p,w1}$ are isomorphic if two levels of sub families are isomorphic. The sub family $K_{v,w}$ converts into a new form by assigning males to all individuals v and females to individuals w. The new sub family is formed by creating new individuals vw_1, vw_2 that are marked as female and the set of edges denoted as
$E(f_1) \leftarrow E(f_1) \cup \{(v, vw), (w, vw)\}$ in $k_{v,w1}$ and the set of edges in new sub family sub(f_2) takes the form
$E(f_2) \leftarrow E(f_2) \cup \{(v, vw), (w, vw)$ in $K_{p,w1}$.

Mapping is applied by using (MCIP) algorithm between f_1 and f_2 to prove the isomorphism as described by Jiang [13]. For all other sub families in F_0 mappings are similar to that of σ.

Let the mapping be σ and let it map the sub families in F_0 to their matching families in F^*. The mapping σ and σ are the same on all other individuals in F_0. The mapping induced by σ on individuals of $F_{0'} = F_0/K_{p,w}$ is an isomorphic mapping from $F_{0'}$ to a $F^{*'} = F^*/K_{p,w}$. Hence, $F_{0'}$ is a sub pedigree of $F^{*'}$ which proves the lemma.

Algorithm PairBipartite and Lemma 2 have helped prove that if there exists a connected component in F^* which is a sub family then a sub family also exists in F_0 and directly determine its mirror image and all other sub family's mirror image in F^*.

3.1.3. Parameterized Algorithms for the BCP Problem

The algorithm Pair BiPartite can be repeatedly applied as long there are sub families in F_0 for which images can be created in F^*, while removing sub families from F_0 and $F^{*'}$ till there exists no more sub families. If F^* is not empty, then F^* constitutes of a path $sp = [u_0, u_1, u_2]$ of length equal to 2. If the collection of sub families exist in F_0 and if F_0 is a sub pedigree of F^*, then there will exist at least one edge among the two edges of sp that will be absent from the isomorphic image from F_0 to F^*. The edge can be removed and F_0 continues to be a sub pedigree of F^*. Having removed the edge in F^*, a branch and search algorithm of time complexity $2^k.n^{O(1)}$ is used, where k is the parameter of the instance (F_0,F^*), since the number of edges decreases by 1. The value k here is equal to number of edges in F^* minus the number of edges in F_0.

http://dx.doi.org/10.21622/ace.2022.02.1.043

To summarize and simplify the discussion, an isomorphism, σ, from F_0 to a sub-pedigree of $F*$, will narrow down to the fact that an edge e in F^* is in σ if the edge e is present in the image of σ. This can further be validated in the consecutive sections.

Lemma 3 *Suppose that (F_0,$F*$) is the instance of (BCP) containing labeled and unlabeled leaves, and the edges be e_1 = [a,b] and e_2 = [b, c], be the two different edges in F^* and they share the common vertex b. If F_0 is a sub pedigree of F^*, then for every isomorphism, σ, from F_0 onto F^*, there exists at least one edge e_1 or e_2 that is not isomorphic or does not have an image in F^*.*

Proof. Let F_0 be a sub pedigree of F^* and two edges e_1 and e_2 be the edges that are in isomorphic mapping from F_0 to a sub pedigree of F^*.

Algorithm Branch-path (F_0,F^*)
Input: (F_0,F^*) is an instance from path covered sub-pedigree
Let u be an individual node that has an out-degree
If $u \leq 1$ in F^* then return;
Elseif $u > 1$;
For all distinct edges e1,e2 in u do
Return Branch-path (F_0,F^*/(E_u/e1,e2)) ;
If u does not exist then stop.

Lemma 4 *Let (F_0,F^*) be an instance of BCP. Let there be a simple path $sp = [u_0, u_1, u_2]$ in F^*, where u_0 is a leaf and u_2 is an individual of degree 2. If F_0, a sub pedigree of F^*, constitutes of labelled and unlabelled leaves, then there exists an isomorphism σ from F_0 onto F^* such that at least one of the edges in the path sp is not mapped by σ*

Proof. Assume that edges $[u_0, u_1], [u_1, u_2]$ are in σ and exhibit isomorphism from F_0 to a sub pedigree of F^*. This is with reference to the proof of lemma 3, which states that no edge of the form $[u_0, z], [z, u_2]$ can be in σ, where z $\neq$ u_2. Since u_2 has an in-degree 2 in F^*, the path $[u_1, u_2]$ is the image of the sub family of the type $K_{p,d}$ under σ in F_0. u_0 is the leaf in F^* and image of the type $K_{1,0}$ in F_0. The isomorphic mapping σ can be modified, such that the sub family of the type $K_{2,2}$ in F_0 be mapped onto $[u_1, u_2]$ in F^* and the sub family of the type $K_{1,0}$ in F_0 be mapped to the leaf u_0 in F^*. It then becomes easy to validate that the final mapping σ' is a σ mapping from F_0 to F^* with the edge $[u_0, u_1]$ not in σ'.

The incorporation of the above lemma has been done using the BCP-I and BCP- II algorithms. The algorithms are described as follows:

In the BCP-I algorithm, an instance of the problem is considered in which F_0 is a set of sub families and F^* is a family. The objective is to find out whether F_0 is a sub pedigree of F^*.

If F^* is null, then F_0 is a sub pedigree of F^*, else if K $<$ 0, then F_0 is not a sub pedigree of F^*. If it contains a single sub family then PairBipartite algorithm will be called to reduce the number of individuals in that sub family, which has same number of individuals in F^* and F_0. Then both families F_0 and F^* are tested for similarities or isomorphism. The other criterion within the PairBipartite can be that $F^{*'}$ contains a simple path of length equal to 2 and $F^{*'}$ can contain labelled and unlabelled leaves where each of the branches will be processed.

http://dx.doi.org/10.21622/ace.2022.02.1.043

Algorithm BCP-I(F_0,F^*,k)
Input : F_0, F^* is an illustration of the BCP problem where F_0 is a set of sub families and F^* is a family.
Output : F_0 is a sub pedigree of F^*

1. If $F_0 = \emptyset$ then Stop //(F^* constitutes the sub pedigree F_0)
2. Else if K < 0 then Stop //(F^* does not constitute the sub pedigree F_0)
3. If F^* contains sub family $K_{p,d}$ then
4. $(F_0',F^{*\prime})$ = PairBipartite(F_0,F^*,$K_{p,d}$); //(k = the difference in the number of edges between the two)
5. Return to BCP-1($F_{0'}$, $F^{*'}$);
6. If F^* is not empty then a simple path $sp = [u_0, u_1, u_2]$ is chosen which is of length 2 then
   a. If u_2 has an out-degree =2 in F^* then
      Branch-Path(F_0,$'$ $F^{*'}$);
      Return BCP-I(F_0,$F^*\backslash$u$_1$,u_2,$k-1$);
   b. Else
      Branch-Path (F_0',$F^{*\prime}$);
      Return BCP-I(F_0,$F^*\backslash$(u$_0$,u_1),[$(k-1)$ or (u_1,u_2)];

Theorem 2 *The algorithm BCP-I solves the problem BCP, by determining F_0 is a sub-pedigree of F^*. The parameter 'k' is the difference between the number of edges in the two sub-pedigrees F_0 and F^* with labelled and unlabeled leaves.*

Proof
Input: (F_0, F^*) as an instance of the BCP problem. Families F_0 and F^* have equal number of individuals. The algorithm terminates execution in case of no individuals are in F^* ($F^* = \emptyset$, for k < 0. Hence, F_0 is not a sub pedigree of F^*.

If F^* contains a sub family $K_{p,d}$ within its family, then according to lemma 3 (F_0' $F^{*'}$) is returned by the algorithm Pair BiPartite(F_0' F^*;$K_{p,d}$) and is solved by recursively calling BCP-I (F_0,F^*,k).

Now, if $F^{*'}$ contains a path $sp = [u_0, u_1, u_2]$ and u_0 is the leaf, then if u_2 in $F^{*'}$ is an individual of degree 2, by lemma 4, if F_0 is a sub pedigree of F^*, then there exists an isomorphism from F_0 to F^* where edge (u_1,u_2) is not in σ.

Therefore if F_0 is a sub pedigree of F^* then F_0 is also a sub pedigree of $F^*\backslash$ (u_1,u_2). If the degree of the individual, say u_2, is larger than 2 then lemma 3 can prove that with every isomorphism mapping σ from F_0 onto F^* either (u_0,u_1) or (u_1,u_2) is not in σ or no edges in the set of edges belonging to individual u_2 is in σ. Thus, if F_0 is a sub pedigree of F^* then the algorithm will return a YES else will return a NO for the instance F_0, F^*.

Theorem 3 *The algorithm BCP-I(F_0, F^*) solves the BCP problem in time O(2.42Kn(k+logn)) where n is the size of an instance which contains labelled and unlabelled leaves. The parameter k represents the difference between the number of edges in the sub pedigree F_0 and F^*.*

http://dx.doi.org/10.21622/ace.2022.02.1.043

Proof

The sub families in F_0 can be organized based on the size of the sub families using appropriate data structures. The largest sub family $K_{v,w}$ in F_0 for a given sub family $K_{p,d}$ in F^* can be found in O(log n) time. If F^* is not empty then from u_0, which is the leaf node in F^*, a path of the form $[u_0,u_1,u_2]$ can be constructed in F^* of length equal to 2.

The Branch and Search Algorithm in BCP-I can be described as a bounded search in which each internal node having more than one child maps to a branch in the algorithm, and the leaf vertex matches to a decision formulated by the algorithm. By removing sub families from F_0 and F^*, a path of length 2 is constructed in F^*. To find out the sub families by removing edges at each step from F^*, the algorithm takes K branching steps, and a time of O(n) to identify the sub families. The algorithm takes time of O(nlogn) for removing the matching sub families.

Assume that, u_2 has a degree ≥ 2 so that, the subset edges E_{u2} contain at least 2 edges, then a computational path can be traced to each leaf. This is achieved using the result of the recurrence relation in You et al. [19], this search results in tracing the path to 2.42K leaves.

Finally, the run time of the algorithm BCP-I is O(2.42K n(k + log n)). This is the time taken to identify sub families and remove them, in addition to the time to search for the computational paths to the leaves when edges are more than 2.

A second version of the algorithm which diminishes the time complexity as per the results reported by [20] is presented as BCP-II. This algorithm executes in $O(bK.n^{O(1)})$ where b is less than 2.42, and n is superimposed by a higher degree polynomial. The BCP-II algorithm is presented in below.

Algorithm BCP-II(F_0, F^*, k)
Input : F_0, F^* is an illustration of the BCP problem where F_0 is a set of sub families and F^* is a family.
Output : F_0 is a sub pedigree of F^*
Parameter: K = $K_0 - K^*$, the difference between the number of edges in F_0 and the number of edges in F^*
If F^* = null then Stop // F^* constitutes the sub pedigree F_0
If k < 0 then Stop// F^* does not constitute the sub pedigree F_0
Select an individual u_1 with the largest out-degree from F^*
If degree(u_1) $\leq c$ then solve by using [20]
If u_1 is a parent P and is in an isolated sub family $K_{p,d}$ then
$(F_0',F^{*'})$ = PairBipartite (F_0, F^*, $K_{u,d}$)
Return BCP-II(F_0', F^*, c, k)
Else select a simple path $sp = [u_0, u_1, u_2]$ of length 2 in F^* then
BranchPath(F_0',F^*)
Return BCP (F_0, $F^*\backslash(u_0,u_1)$, c, $k-1$)
OR BCP (F_0, $F^*\backslash(u_1,u_2)$, c, $k-1$)
OR BCP (F_0, $F^*\backslash Eu_1$,c,k-1,Eu_1);

Associated with the above algorithm is the theorem and the subsequent proof that algorithm BCP-II can be executed in a time that is upper bounded by $O(bK.n^{O(1)})$ where c is a constant ≥ 1, n is the size of the sub pedigree instance (F_0,F^*) constituting of labelled and unlabelled leaves, and k denotes the difference of number of edges.

http://dx.doi.org/10.21622/ace.2022.02.1.043

Theorem 4 *The algorithm BCP-II (F_0,F^*,c,k) solves the problem BCP in time O(bK $n^{O(1)}$), when there exists a constant c ≥ 1, and n is the size of the instance (F_0,F^*) with labelled and unlabeled leaves, and k denotes difference in the number of edges between sub pedigrees F_0,F^*.*

Proof

The algorithm BCP-I, undergoes changes with the inclusion of step 4, as denoted in the preceding BCP-II algorithm, which requires validation and justification. In a study conducted by Baumbach et al. [20], they discussed (Star Cover Tree) SCT problem which constitutes of a tree and a set of stars. These stars can be connected through edges between them in such a way that the resulting tree is isomorphic to T. Baumbach et al. [20] have proved that when the count of unique stars is bounded by a constant k, CTS can be solved in polynomial time of order $O(n^{2c+3})$. In this work, simple undirected graphs are considered. The star constitutes of a set of vertices connected by edges that define the size with at least one internal vertex.

The following is built on Theorem 2 in [14], and the fact that the bipartite graph consists of two equal sized stars, such that one of the two stars is present on the path. Any sub-pedigree with labelled and unlabeled leaves represents a branch of the sub-pedigree which has a connected component of distinct sub-families, so that the Tree Edit Distance with insertion and deletion of edges can be used to solve the BCP problem, where c represents various sizes of sub-families in F_0.

Moreover, if the sub-families in F_0 have size bounded by c, then F_0 has at most (c + 1) sub-families of various types inclusive of sub-families without children having a size of 0. Therefore, the BCP problem can be solved in time of $O(n^{2c+5})$. When all the connected components of sub families of size (c + 2) in F^* are linked, such that the sub-families form a new sub pedigree graph with a definite number of sub-families, and have the same size in F_0, are added on to F^*, then the algorithm executes in time of $O(n^{2c+9})$, where the sub-families in F_0 have their size bounded by c.

If an individual has a degree ofr edge $(u_1) \leq c$, in order to let F_0 be a sub-pedigree of F^*, no sub-family in F_0 can have size exceeding c because u_1 is the individual with the maximum out degree in F^*. Therefore, from the discussion above this step is solved in time $O(n^{2c+9}) = n^{O(1)}$, where the polynomial component is the complexity of higher degree. If u_1 is not an isolated sub-family in F^*, then there exists a simple path of length 2.

Applying lemma 4, the complexity for each computational path in the search takes polynomial time, again the branching of the algorithm gives the recurrence relation P(X) = 2P(X − 1) + P(X–c), where P(X) defines the number of leaves.

Using the standard method shown by Chen et al. [21], the above recurrence relation is solved to obtain $P(x) = x^c - 2x^{c-1} - 1$, with root b > 2, and the total number of leaves in the search is bK. Thus, the algorithm BCP-II is executed in $b^k n^{O(1)}$.

Next the concept of Kernelization is introduced and it is proved that linear time kernelization algorithm yields a kernel of size $O(K^3)$ for the BCP problem. The theorem to prove the performance of the algorithm follows.

Theorem 5 *The problem "A set of Bipartite graphs Covering the Pedigree graphs (BCP)" is executed in time $O((2 + EPSILON)^k n^{O(1)})$, where the constant EPSILON ¿ 0.*

http://dx.doi.org/10.21622/ace.2022.02.1.043

Proof

The proof depends on theorems 3 and 4 for solving the algorithms BCP-I and BCP-II, when EPSILON ¿ 0. Given an instance (F_0, P) of the BCP problem, a reduction rule is applied by the Kernelization algorithm which removes certain leaves from F_0 and P such that there exists a sub family in F_0 whose size is bounded by O(K^2). It is said that a K vertex cover has a kernel of size O(K^2).

K vertex cover algorithm tries to identify whether a set of vertices in a graph covers all the edges of the graph or not. As a subsequence, the Kernelization algorithm takes a space which is upper bounded by O(K^3), that is the size of an instance (F_0', P′) of BCP, where the total number of individuals are bounded by O(K^3), that is the time taken to identify and remove the sub families and to identify the branch path till the leaf node. Proof of the above algorithm is done using the method of construction in lemma 5.

Lemma 5 *Suppose that (F_0, P) is an instance of BCP. Let z be any individual with an out-degree larger than k + 1 in the sub-pedigree F^*. Then there exists an isomorphism from F_0 to a sub-pedigree of P, where the individual z is the image of the parent of a sub-family in F_0. Moreover, at least degree(z)–k + 1 neighbours of z are leaves in F^*.*

Proof

Let σ be an isomorphism from F_0 to a sub pedigree P, such that that there exists a set E_σ in F^* that cannot be in the set E_σ, but if z is the image of a leaf then it is bounded by 1 only. Also z has a degree $\geq K+1$ in F^* which contradicts the assumption that E_σ has only k edges. Hence, z has to be the image of the internal parent of a sub family in F_0.

Kernel BCP (F_0,P,$K_{m,n}$)

Input: F_0,P is an instance of BCP
// b is the set of sub families in F_0
Output: (F_0',P')
If the number of individuals in P = 0;
Then stop NO-instance of BCP;
b = Sub-family Algorithm(F_0,$K_{p,w}$);
T = Sub-family Algorithm (F^*,$K_{u,x}$);
If | b | ≠ | T | then stop;
F_0 is not a sub-pedigree of P
Order the sub-families in F_0 by ascending order that makes a gap;
For each two adjacent sub-families $K_{p,d}$ and $K_{p,w}$ that d−w > k + 1 do
Arbitrarily remove $d-w+K+1$) leaf individuals from each sub-family in b and T;
If the degree of the smallest sub-family in F_0 is greater than K+2 then
Return Kernel-BCP (F_0', P',$K_{m,n}$)
Return (F_0,P)

The kernelization algorithm for BCP is applied repeatedly, whenever there is a sub-family in F_0 that can be removed. At the end of the process, there are subfamilies that cannot be removed and the researchers obtain equal instances (F_0', P′) of BCP.

Lemma 6 *Suppose that the algorithm Kernel-BCP returns an instance (F_0',P') which are sub pedigrees of F_0 and P. F_0 is a sub pedigree if and only if F_0' is a sub pedigree of P'.*

http://dx.doi.org/10.21622/ace.2022.02.1.043

Proof

The families F_0 and P must have the same number of individuals. If P = Øor the number of sub families in $F_0 \neq$ P then there is no instance of the BCP problem. Step 1 and Step 4 of the algorithm handle these cases appropriately.

Assume that sub families in F_0 are ordered by the ascending order and there are two adjacent sub families $K_{p,d}$ and $K_{v,w}$ (w < d) in F_0 where d – w > k + 1, respectively. Images of the sub family in F_0 are found in P directly and some being larger than K_p, d are redundant and their images are removed. This process is used to reduce the scale of F_0 and P for an isomorphism from F_0 to P.

Theorem 6 *The Kernel-BCP algorithm is a linear time algorithm that returns an instance (F_0', P') for the BCP problem, and the number of individuals in a new instance is bounded by O (K^3), where k represents the count of sub-families in F_0.*

Proof

Since the Kernel-BCP algorithm is repeatedly applied whenever there is a sub-family in an ascending order in F_0 that creates a gap when removed. Therefore, all sub-families that can be removed from F_0 are removed in a single scanning in an ascending order in F_0. Once a sub-family which creates a gap is identified, a set of leaves in F_0 and P can be removed in linear time. Then, size of each sub-family in F_0 is not larger than n (n = the size of the sub-families) and k represents the number of sub-families. The sorting of the sub-families can be done in O(kn) time, which can be reduced to O(n) by using Counting-Sort. From the above discussion and lemma (6) and (5), an equivalent instance (F_0',P′), the total count of individuals in F_0' and P′ is bounded by O(K^3).

4. CONCLUSIONS AND FUTURE WORK

The study can extend towards the evaluation of reconstructed pedigree graphs until the third generation limited to the families and sub-families and also as a generalized algorithm until the seventh generation.

The non-availability of the requisite datasets acted as a limitation to the practical verification of the study. Although, from the theoretical viewpoint the algorithms showed promising results.

The evaluation of a large pedigree graph with labeled and unlabeled leaves, and proving isomorphism with the real pedigree is a problem that has been addressed critically and the solution has been presented to the geneticist for use. The paper introduces the problem of comparing labeled and unlabeled pedigree graphs. It also presents the parameterized and kernelization algorithms. The parameterized algorithms are supported by the standard branch-and-search process. Kernelization algorithms have a complexity of $O(k^3)$ for BCP. For further progress in research, the model presented can be generalized to allow the evaluation of unlabeled sub-pedigree with a large number of individuals and sub-families up to seven generations. Furthermore, experimental results are needed to test the applicability of the proposed algorithms. The evaluation of a largely reconstructed pedigree with labeled and unlabeled leaves and establishing isomorphism with the original shall help in unearthing many genetic facts with regards to the diseases.

http://dx.doi.org/10.21622/ace.2022.02.1.043

The research compares the labeled and unlabeled pedigrees and sub pedigrees with the original and uses the parameterized and kernelization algorithms to generate promising results as compared to theoretical study. Here, the kernelization algorithm utilizes the concept of reducing the size of the input as a function of the parameter, thereby removing the sub-pedigrees and sub-families to prove the isomorphic attribute with the original. The future scope of work is to incorporate the generalization of algorithms to accommodate multiple generations of a pedigree.

REFERENCES

[1] D. He, Z. Wang, E. Eskin, "Iped2: Inheritance path based pedigree reconstruction algorithm for complicated pedigrees," IEEE/ACM Trans. Comput. Biol. Bioinformatics, Vol.14, issue no 5, pp.1094-1103, September 2017.

[2] M. Ghimiray, R . Vernooy."The importance and challenges of crop germplasm interdependence: the case of Bhutan," Food Secur. Vol. 9, pp.301-310. April 2017.

[3] TY. Berger-Wolf, SI. Sheikh, et al., "Reconstructing sibling relationships in wild populations," Bioinformatics, Vol. 23, pp.49-56. 2007.

[4] Brown, DG.; Berger-Wolf, T. Discovering kinship through small subsets, In Algorithms in Bioinformatics; Moulton, V., Singh, M., Eds.; Springer:Berlin, Heidelberg, 2010; pp. 111-123.

[5] Thompson, EA. In Pedigree Analysis in Human Genetics ;Baltimore: Johns Hop-Kins University Press, 1986.

[6] B. Kirkpatrick, SC. Li, RM. Karp, E. Halperin. "Pedigree reconstruction using identity by descent," J. Comput. Biol. Vol.18, issue no 11, pp.136-152. November 2011.

[7] L. Sun, K. Wilder, MS. McPeek. "Enhanced pedigree error detection," Hum Hered. Vol.54, issue no 2, pp. 99-110. October 2002.

[8] MS. McPeek, L. Sun. "Statistical tests for detection of misspecied relationships by use of genome screen data," Am. J. Hum. Genet. Vol.66, issue no 3, pp.1076-1094. March 2000.

[9] M. Boehnk, NJ. Cox. "Accurate inference of relationships in sib-pair linkage studies," Am. J. Hum. Genet. Vol.61, issue no 2, pp. 423-429. August 1997.

[10] B. Kirkpatrick, Y. Reshef, H. Finucane, H. Jiang, B. Zhu, RM. Karp. "Comparing pedigree graphs," J. Comput. Biol. Vol.19, issue no 9, pp. 998-1014. October 2012.

[11] Z. Chen, Q. Feng, C. Shen, J. Wang, L. Wang. "Algorithms for pedigree comparison," IEEE/ACM Trans. Comput. Biology Bioinform, Vol.15, issue no 2, pp. 422-431. April 2018.

http://dx.doi.org/10.21622/ace.2022.02.1.043

[12] CM. Herbinger, P. TO'Reilly, RW. Doyle, JM. Wright, F. O'Flynn. "Early growth performance of Atlantic salmon full-sib families reared in single family tanks versus in mixed family tanks," Aquaculture, Vol.173, issue no 14, pp. 105-116. March 1999.

[13] H. Jiang, G. Lin, W. Tong, D. Zhu, B.Zhu. "Isomorphism and similarity for 2-generation pedigrees," BMC Bioinformatics, Vol.16, issue no Suppl(5), pp. 1-8. March 2015.

[14] LA. Amar, NA. Belal, S. Rashwan. "Comparing unlabeled pedigree graphs via covering with bipartite and pat," J. Comput. Biol. Vol.23, issue no 11, pp. 912-922. November 2016.

[15] Braun, Bremen L and Schott, David A and Portwood, John L and Andorf, Carson M and Sen, Taner Z. "PedigreeNet: a web-based pedigree viewer for biological databases," Bioinformatics, 35(20), pp. 4184-4186 October 2019.

[16] N. Xie, Q. Sun, J. Yang, Y. Zhou, H. Xu, L. Zhou, Y. Zhou. "High clinical heterogeneity in a Chinese pedigree of retinal vasculopathy with cerebral leukoencephalopathy and systemic manifestations (RVCL-S)," Orphanet Journal of Rare Diseases,16(1), pp.1-15, Jan 2021.

[17] H. Yu, G.Morota. "GCA: an R package for genetic connectedness analysis using pedigree and genomic data," BMC Genomics, 22(119), pp.1-8, February 2021.

[18] Z.J. Rawandoozi, T.P. Hartmann, S. Carpenedo, K. Gasic, et al., "Mapping and characterization QTLs for phenological traits in seven pedigree-connected peach families," BMC Genomics, 22(187), pp.1-16, March 2021.

[19] J. You, J. Wang, Q. Feng, F. Shi. "Kernelization and parameterized algorithms for covering a tree by a set of stars or paths," Theor. Comput. Sci. Vol 607, part 2, pp. 257-270. November 2015.

[20] J. Baumbach, J.Guo, R. Ibragimov. "Covering tree with stars," In: Computing and Combinatorics.;19th International Conference; COCOON; Springer: Hangzhou, China, 2013.

[21] J. Chen, IA. Kanj, W. Jia. "Vertex cover: Further observations and further improvements," J. Algorithms, Vol.41, issue no 2, pp. 280-301. November 2001.

http://dx.doi.org/10.21622/ace.2022.02.1.058

MEMS and NEMS - Micro (and Nano) Electromechanical Systems

Roshdy A. Abdelrassoul

College of Engineering and Technology, Arab Academy for Science, Technology and Maritime Transport

Email: {roshdy@ieee.org}

Received on, 28 May 2022 - Accepted on, 18 June 2022 - Published on, 30 June 2022

ABSTRACT

MEMS and NEMS use and application are growing and their market is expanding. With the development of 5G and IoT technologies, the number of MEMS components is always increasing.

In this paper gives an introduction of MEMS/NEMS definition, categories, and advantages. Fabrication techniques of MEMS/NEMS are also discussed. Finally, an elaborative analysis of the different applications of MEMS/NEMS is highlighted, followed by a discussion of the future of Micro/Nano electromechanical Systems.

Index Terms: MEMS; NEMS; Micro Electromechanical Systems; Nano Electromechanical Systems

1. INTRODUCTION

MEMS – Micro electromechanical Systems, aka MST - Microsystems Technologies in Europe, are manmade devices created by using compatible micro fabrication techniques that are capable of converting physical stimuli, events and parameters to electrical, mechanical and optical signals, as shown in Figure 1, performing actuation, sensing and other functions.

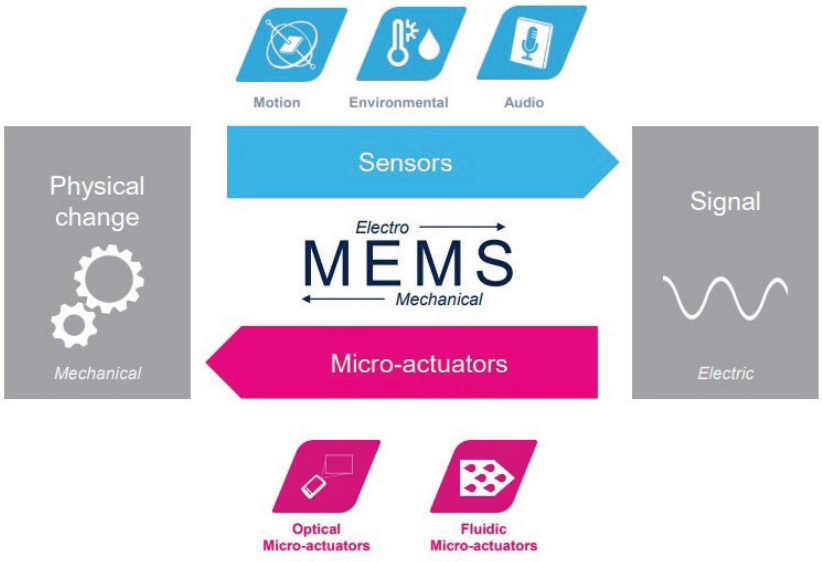


Fig. 1. MEMS sensors and actuators [1]

MEMS/NEMS are fabricated using micromachining technology, and are used for sensing, actuation or are passive micro-structures. They are usually integrated with electronic circuitry for control and/or information processing.

http://dx.doi.org/10.21622/ace.2022.02.1.058

MEMS/NEMS range in size from a millionth of a meter (micrometer) to a thousandth of a meter (millimeter). Figure 2 shows the scale of things down to nanometer scale. They are manufactured onto semiconductor material (Si, silicon carbide, metals, plastics), integrating various sensing, computing and actuating elements into a single chip. Figure 3 shows a scanning electron microscope (SEM) photo of a part of a MEMS device under microscope. The size of the gear is 10 micrometers.

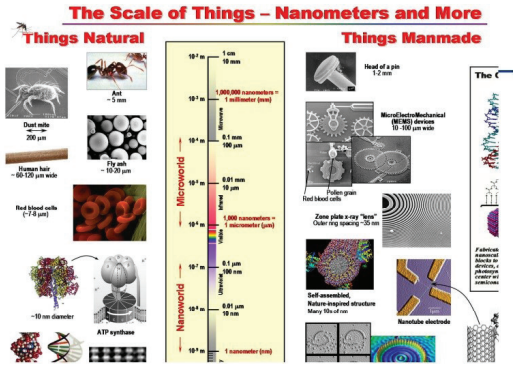


Fig. 2. The scale of things; Natural and Manmade assembly. [2]

Fig. 3. Part of a MEMS device under microscope. [3]

Figure 4 shows a timeline of key microsystem developments, showing the start of MEMS commercial products in 1977.

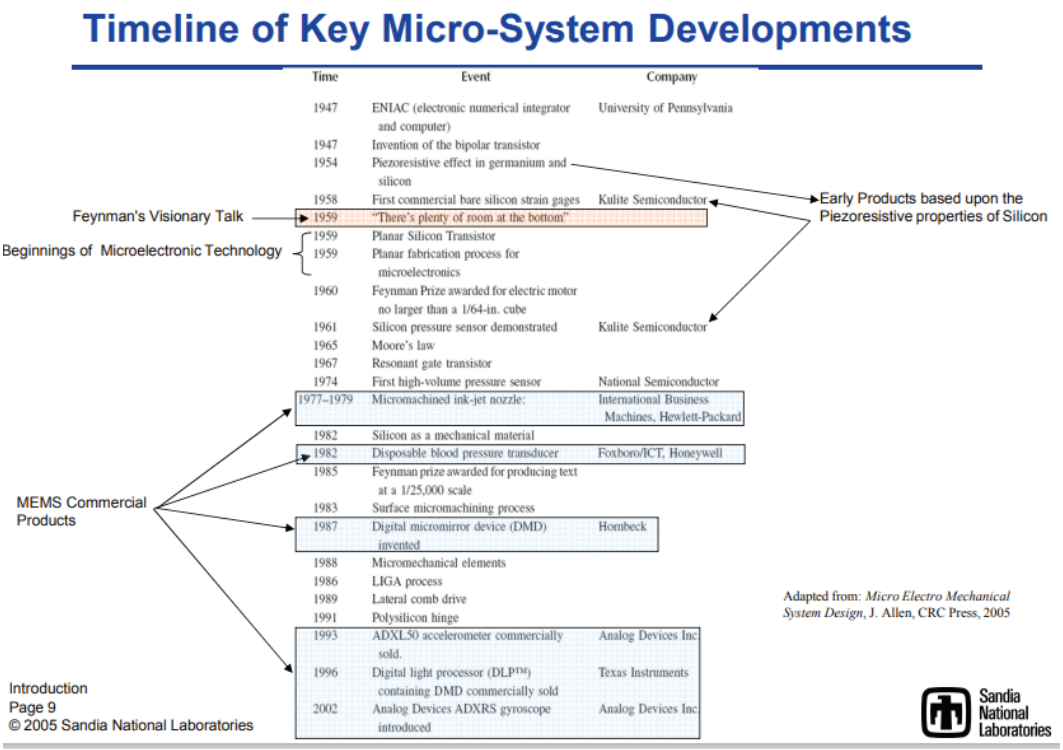


Fig. 4. Timeline of key microsystem developments [4]

2. ADVANTAGES OF MEMS/NEMS

MEMS/NEMS advantages include, miniaturized size, high sensitivity, low noise, reduced cost, and batch processing.
MEMS can be categorized into four categories: Fluidic, RF, optical, and Bio-MEMS, as shown in Figure 5.

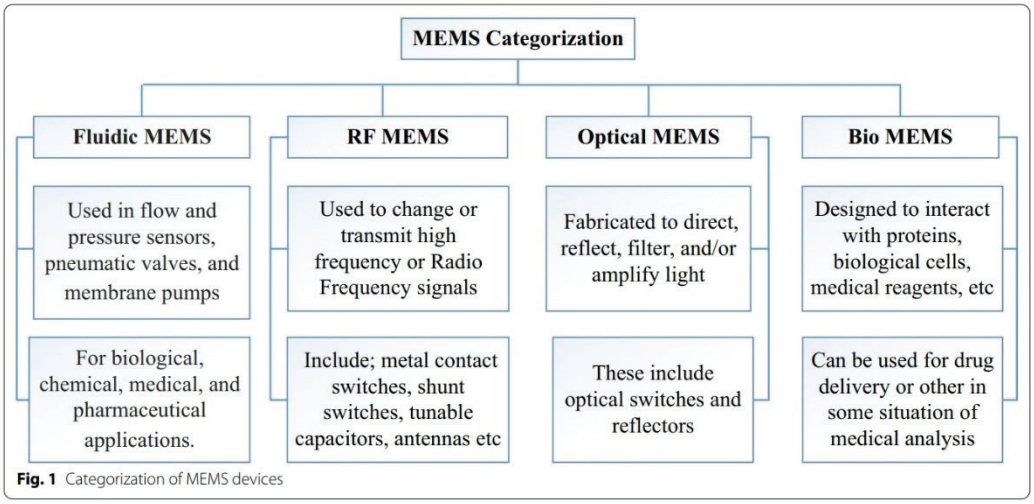


Fig. 5. MEMS categories [5]

MEMS sensors and actuators are based on capacitive, piezoelectric/piezoresistive, electrothermal, electrostatic, electromagnetic, and optical mechanisms, as shown in Figure 6.

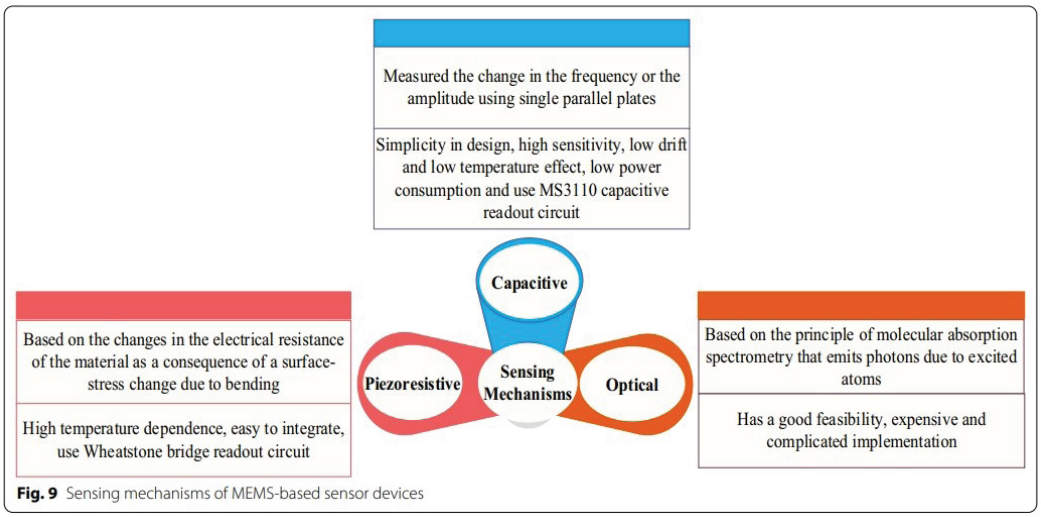


Fig. 6.a. Sensing mechanisms of MEMS based sensors [5]

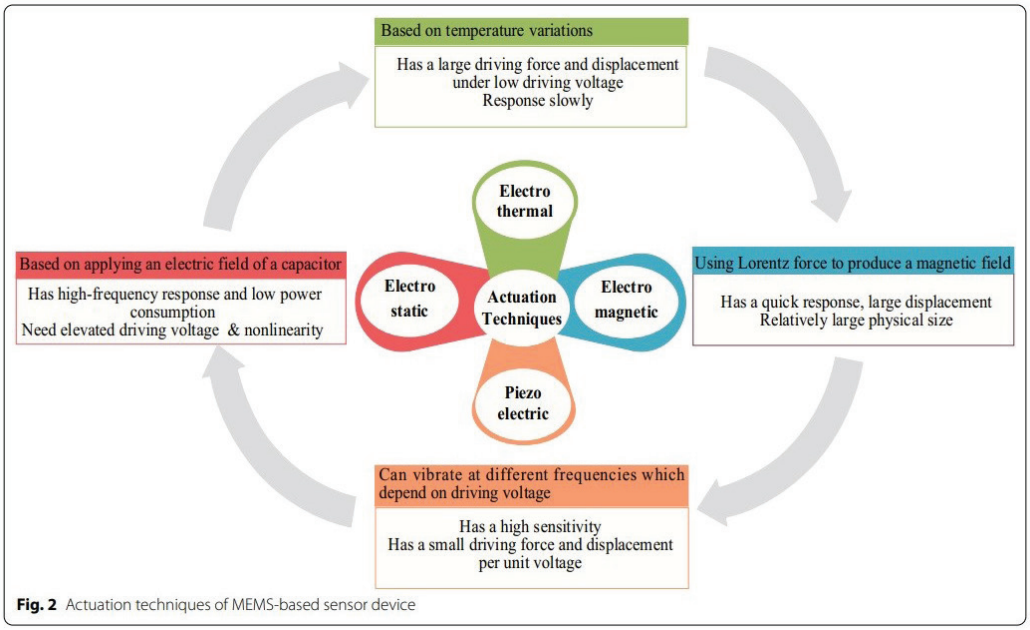


Fig. 6.b. Actuating mechanisms of MEMS based sensors [5]

http://dx.doi.org/10.21622/ace.2022.02.1.058

Table 1 shows the different energy domains handled in MEMS.

Table 1
Different energy domains handled in MEMS [5]

Nature	Chemical domain	Electrical domain	Mechanical domain	Thermal domain	Radiative domain	Magnetic domain
Potential	Chemical concentration	Voltage	Force	Temperature	Electromagnetic waves	Magnetic field strength
Flow	Reaction rate	Current	Velocity	Entropy flow rate	Infrared radiation	Magnetic direction
Generalized displacement	Molecule recognition	Charge	Displacement	Entropy	Transmission	Electromagnetic force
Generalized resistance	DNA sequence	Resistance	Damping	Thermal resistance	UV radiation	Lorentz force
Generalized inductance	DNA hybridization	Inductance	Mass	–	X-rays	Induction
Generalized capacitance	Protein construct	Capacitance	Mechanical compliance	Thermal capacitance	Absorption	–

3. MEMS/NEMS FABRICATION

MEMS/NEMS fabrication can be divided into surface micromachining and bulk micromachining. A third process was developed specifically for MEMS and is called LIGA, as shown in Figure 7.

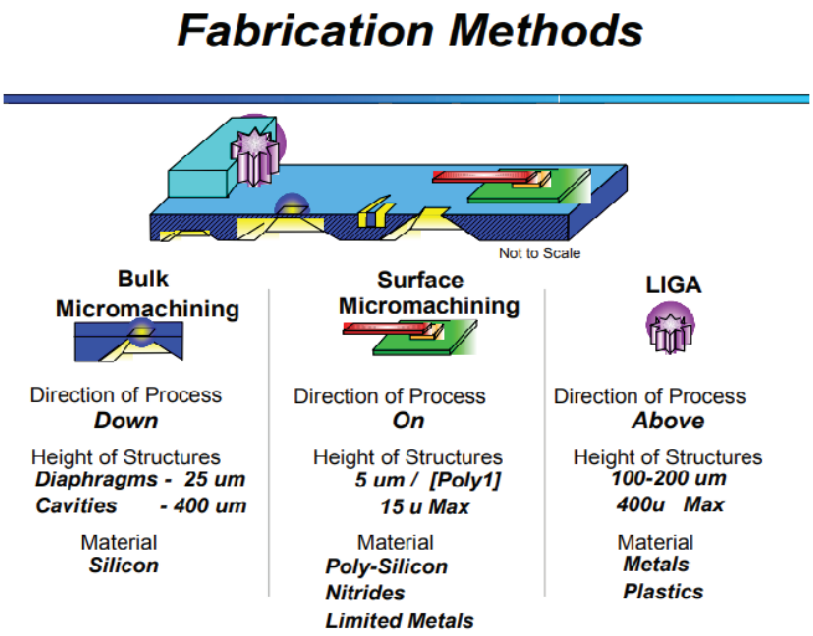


Fig. 7. Bulk-, surface micromachining and LIGA. [6]

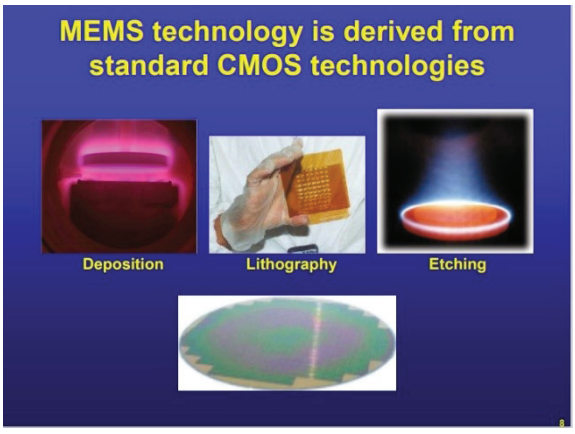


Fig. 8. MEMS fabrication processes. [7]

http://dx.doi.org/10.21622/ace.2022.02.1.058

The three basic fabrication processes used in integrated circuit fabrication are also the basic processes in MEMS/NEMS fabrication, namely, Deposition, Patterning, and Etching, as shown in Figure 8.

DEPOSITION is the addition of layers of materials, including oxidation, epitaxy, evaporation, sputtering, spin-on methods, chemical-vapor deposition (CVD/PECVD/LPCVD), diffusion, and ion implantation.

PATTERNING is done using Photolithography, electron-beam (e-Beam) lithography, and X-ray lithography.

ETCHING is removal of material by wet etching (Isotropic), and dry etching (anisotropic), including plasma etching and reactive ion etching (RIE), and deep reactive ion etching (DRIE).

4. APPLICATIONS OF MICRO/NANO ELECTROMECHANICAL SYSTEMS (MEMS / NEMS)

MEMS/NEMS have a lot of applications and are found almost everywhere around us, as shown in Figure 9, with a market share of $7.8B in 2008, growing at about 14% every year.

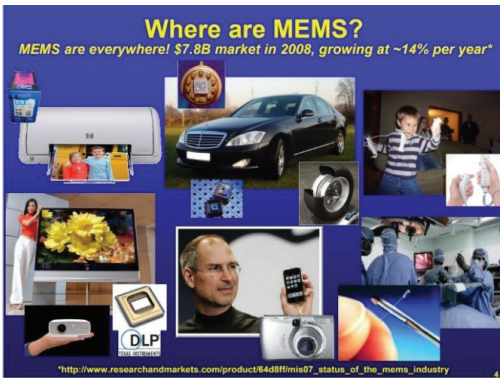


Fig. 9. MEMS are everywhere. [7]

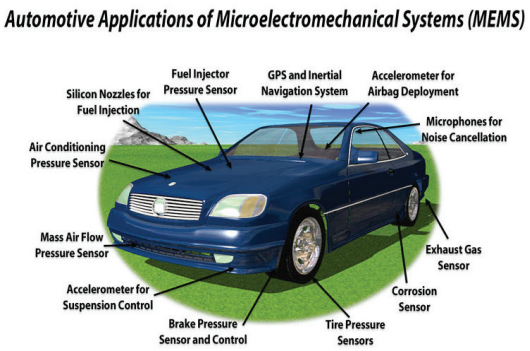


Fig. 10. Automotive applications of MEMS/NEMS. [8], [9]

MEMS/NEMS applications include AUTOMOTIVE APPLICATIONS, as shown in Figure 10, like sensors, accelerometers (Figure 11) – air bag crash sensing – seat belt tension – automobile suspension control, vibration – engine management – security devices, angle of inclination, vehicle stability and roll.

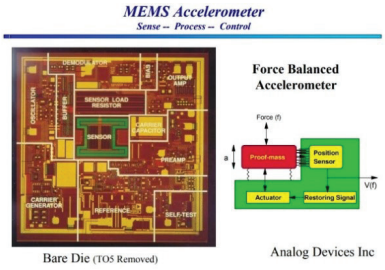


Fig. 11. MEMS accelerometer. [6]

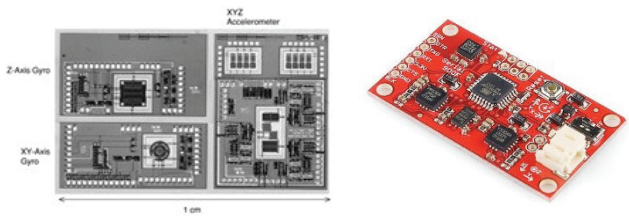


Fig. 12. MEMS inertial measurement unit (IMU). [10], [11]

http://dx.doi.org/10.21622/ace.2022.02.1.058

Fig. 13. MEMS gyroscope. [6]

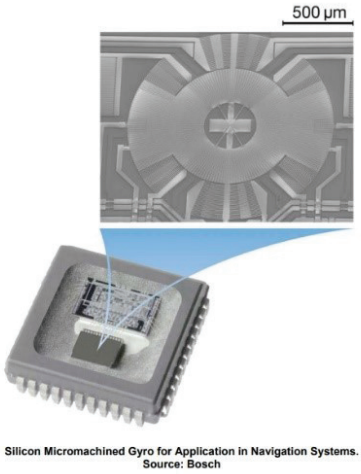


Fig. 14. MEMS gyro used for navigation systems. [6]

BIOLOGICAL APPLICATIONS, including microfluidics, Lab-on-a-Chip, micropumps, resonant microbalances, and Micro Total Analysis systems,
WIRELESS COMMUNICATIONS, including micromechanical resonator for resonant circuits and filters, and RF switches, inertial measurement unit (IMU) shown in Figure 12,
OPTICAL COMMUNICATIONS, including optical switching,
TRANSPORTATION, including gyroscopes (Figure 13),
AEROSPACE, including gyroscopes (Figure 14), deformable mirrors first developed in the 1990's (Figure 15),

Fig. 15. MEMS deformable mirror. [12]

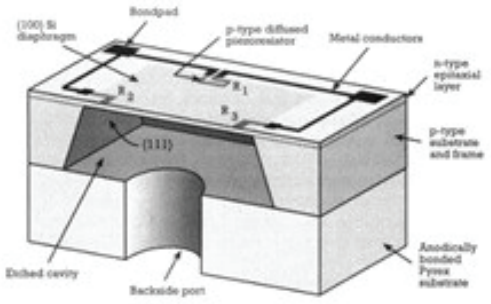


Fig. 16. MEMS Piezoresistive pressure sensor. [2]

HEALTHCARE, including blood pressure sensor, human activity for pacemaker control, wearable health monitoring,
INDUSTRY APPLICATIONS, including pressure sensors (Figure 16), manifold absolute pressure (MAP) sensor,
COMPUTING, including inkjet print heads (Figure 17), magnetic read/write heads for hard disks, Digital Micromirror Device (DMD) used in projection systems (Figure 18).

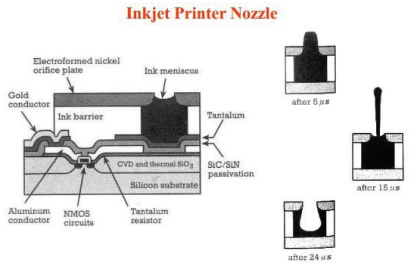
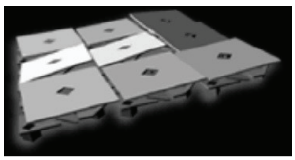


Fig. 17. MEMS inkjet print head. [2]



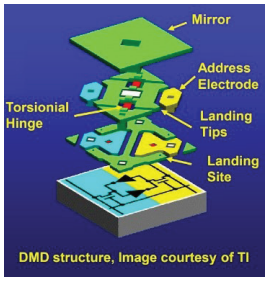


Fig. 18. MEMS Digital Micromirror Device (DMD). [7]

http://dx.doi.org/10.21622/ace.2022.02.1.058

ENVIRONMENTAL APPLICATIONS, including air quality monitoring, toxic gas detection, monitoring of seismic activity, RFID systems (Figure 19), and energy harvesting (Figure 20), using electromagnetic energy, acoustical energy and vibrational energy.

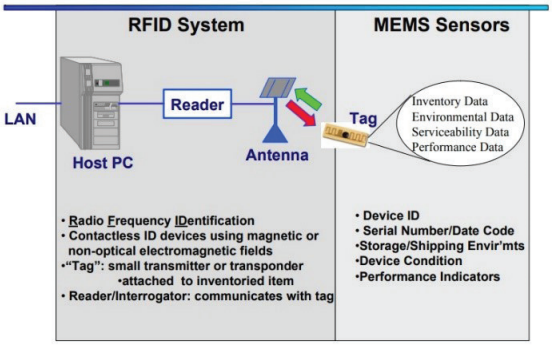


Fig. 19. MEMS RFID system. [6]

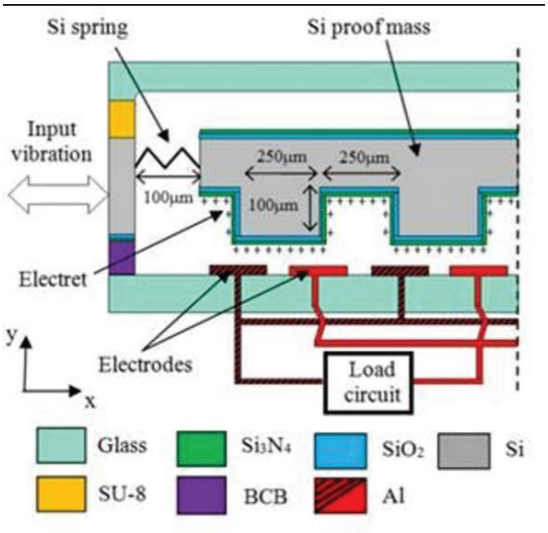


Fig. 20. MEMS energy harvesting system. [13,14]

5. FUTURE OF MICRO / NANO ELECTROMECHANICAL SYSTEMS (MEMS/NEMS)

Figure 21 shows the number of mechanical components versus the number of transistors for different MEMS applications, showing the location of the majority of existing MEMS systems.
Figure 22 shows the development trends and perspectives of future MEMS/NEMS systems.

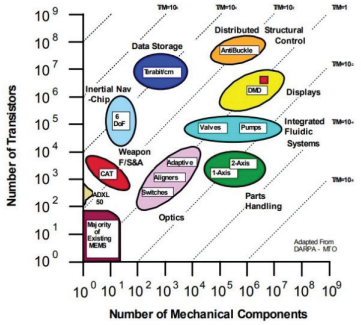


Fig. 21. Number of mechanical components vs number of transistors for different MEMS applications. [6]

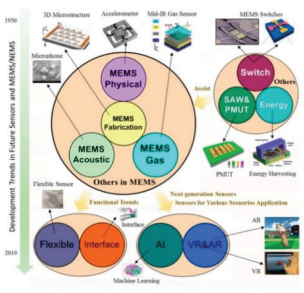


Figure 1. The development trends and perspectives of the future sensors and microelectromechanical systems (MEMS)/NEMS.

Fig. 22. Development trends of future MEMS/NEMS systems. [15]

http://dx.doi.org/10.21622/ace.2022.02.1.058

MEMS market is growing and expanding, as is evident from Figure 23, which shows the 2020 Top MEMS manufacturers, and Figure 24 which shows the MEMS market Forecast.

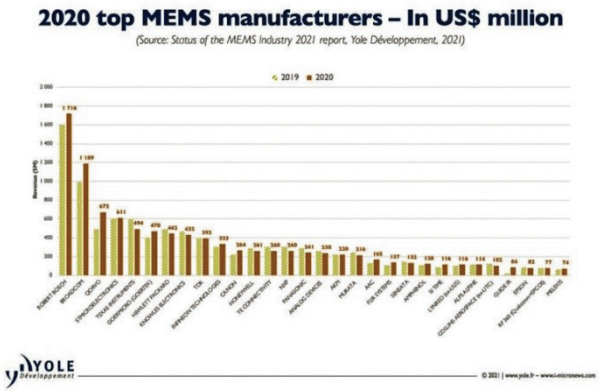


Fig. 23. 2020 Top MEMS manufacturers. [15]

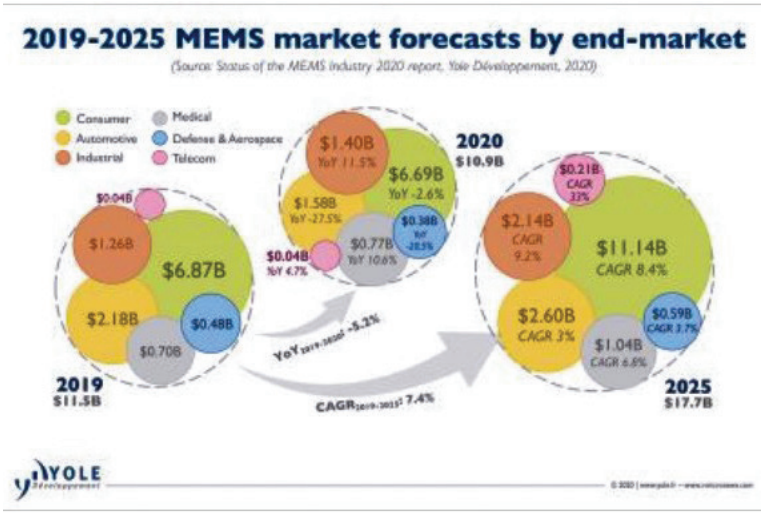


Fig. 24. MEMS market Forecast. [15]

REFERENCES

[1] A. Picco, "Materials, Sensors and Actuators in MEMS technology evolution," in 2020 Nano Innovation Conference & Exhibition, Rome, 15-18 Sept. 2020.

[2] J. Lin, "Microelectromechanical Systems (MEMS) An introduction," ppt [Online], 2015. Available: https://ppt-online.org/416691.

[3] D. Raja "An Overview of Various MEMS Devices and their Applications," Circuit Digest Tutorial, December 11, 2019. Available: https://circuitdigest.com/tutorial/what-is-mems-various-mems-devices-and-applications.

[4] J. Allen, Micro electro mechanical system design, 1st ed., Boca Raton: CRC Press, 2005.

[5] A. Algamili, M. Khir, J. Dennis, A. Ahmed, S. Alabsi, S. Hashwan and M. Junaid, "A Review of Actuation and Sensing Mechanisms in MEMS-Based Sensor Devices," Nanoscale Res Lett, vol. 16, issue. 16, pp. 1-21, 26 January 2021.

http://dx.doi.org/10.21622/ace.2022.02.1.058

[6] J. Zhu, X. Liu, Q. Shi, T. He, Z. Sun, X. Guo, W. Liu, O. Sulaiman, B. Dong and C. Lee, "Development Trends and Perspectives of Future Sensors and MEMS/NEMS," Micromachines, vol. 11, issue. 7, pp. 1-30, Jan. 2020.

[7] J. Walraven, "Introduction to Microelectromechanical Systems (MEMS) Materials, Fabrication Processes and Failure Analysis," in 38th International Symposium for Testing and Failure Analysis, American Society for Metals, Nov. 2012.

[8] G. Badilla, B. Salas, M. Wiener and C. González, "Microscopy and Spectroscopy Analysis of Mems Corrosion Used in the Electronics Industry of the Baja California Region, Mexico," in Air Quality, G. Badilla, B. Valdez and M. Schorr, Eds. IntechOpen: 2012, pp. 163-184.

[9] I-MicroNEWS, "MEMS industry: the headwinds from COVID-19 and the way forward," Extracted from: Yole Development, "Status of the MEMS Industry, 2020," Available: chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://s3.i-micronews.com/uploads/2020/07/YDR20104-Status-of-the-MEMS-Industry-2020-Sample-Yole-D%C3%A9veloppement.pdf.

[10] Sparkfun, "QuadroCopter: Inertial measurement unit (IMU). 9 Degrees of Freedom - Razor IMU - AHRS compatible," SEN-10125 ROHS datasheet, 2010.

[11] A. El-Fatatry, "Inertial Measurement Units – IMU," presented at the RTO AVT Lecture Series on "MEMS Aerospace Applications," Paper presented at the RTO AVT Lecture Series on "MEMS Aerospace Applications", held in Montreal, Canada, 3-4 October 2002; Ankara, Turkey, 24-25 February 2003; Brussels, Belgium, 27-28 February 2003; Monterey, CA, USA, 3-4 March 2003, and published in RTO-EN-AVT-105.

[12] R. Morgan, E. Douglas, G. Allan, P. Bierden, S. Chakrabarti, T. Cook, M. Egan, G. Furesz, J. Gubner, Tyler D. Groff, Christian A. Haughwout, Bobby G. Holden, C. Mendillo, M. Ouellet, P. Pereira, A. Stein, S. Thibault, X. Wu, Y. Xin and K. Cahoy, "MEMS Deformable Mirrors for Space-BasedHigh-Contrast Imaging," Micromachines, vol. 10, issue 366, pp. 1-20, 2019.

[13] IDTechEx, "High-performance electrostatic MEMS vibration energy harvester," Availabe: High-performance electrostatic MEMS vibration energy harvester | Off Grid Energy Independence

[14] R. Das, "Energy Harvesting and Storage 2014-2024: Forecasts, Technologies, and Players," Available: Energy Harvesting and Storage 2014-2024: Forecasts, Technologies, Players: IDTechEx

[15] i-MicroNEWS, "A brave new MEMS world: a $18.2B market by 2026," Available: http://www.yole.fr/

http://dx.doi.org/10.21622/ace.2022.02.1.058

BIOGRAPHY

Roshdy A. AbdelRassoul is a Professor Emeritus at the Arab Academy for Science & Technology, Alexandria. Dr. Roshdy received his B.Sc. degree (with Highest Honors) in Electrical Engineering (Communication Engineering) and his M.Sc. degree in Electrical Engineering from Alexandria University, Egypt, in 1973 and 1976, respectively. He received his Ph.D. in Electrical Engineering from Southern Methodist University (SMU), Dallas, TX, USA, in 1981. He then worked as an Assistant Professor in Louisiana State University, Baton Rouge, LA, USA, from 1981 to 1982, after which he joined the Electronics and Communication Engineering Department at Mansoura University, Egypt. Dr. Roshdy worked also as an Assistant Professor at Southern University, Baton Rouge, LA, USA, from 1985 to 1989, before returning to Mansoura University, Egypt. From 1994 to 1995 Dr. Roshdy was a visiting Associate Professor in the Electrical Engineering Department at the Arizona State University, Tempe, AZ, USA. Between 1995 and 1999 Dr. Roshdy was an Associate Professor at King Saud University, Riyadh, Saudi Arabia.

In 2002, he joined the Arab Academy for Science and Technology, Alexandria, Egypt, as a Professor in the Electronics and Communication Engineering Department. Between 2008 and 2010, he was a Professor at Qassim University, Bureidah, Saudi Arabia.

Dr. Roshdy's current research interests include Nanoelectronics, VLSI, Integrated circuit fabrication, analog and mixed-circuits, semiconductor devices, and photovoltaics.