

KELM-KPCA Method for COVID-19-induced Pneumonia Detection

¹BACHA SAWSSEN, ²TAOUALI OKBA, ¹LIOUANE NOUREDDINE

¹Laboratory of Automatic, Signal and Image Processing, National Engineering School of Monastir,
University of Monastir, TUNISIA

²Department of Computer Engineering, Faculty of Computers and Information Technology,
University of Tabuk, Tabuk, SAUDI ARABIA

Abstract— The new corona virus 2019 (COVID-19) has become the most pressing issue facing mankind. Like a wildfire burning through the world, the COVID-19 disease has changed the global landscape in only one year. In this mini-review, a novel image classifier based on Kernel Extreme Learning Machine (KELM) and Kernel Principal Component Analysis (KPCA) is presented. The proposed algorithm called KELM-KPCA, aims to detect COVID-19 disease in chest radiographs, using a constrained dataset.

Keywords— coronavirus; COVID-19; classification; chest CT scans; machine learning

Received: May 11, 2020. Revised: October 1, 2020. Accepted: October 23, 2020. Published: November 5, 2020.

1. Introduction

COVID-19 is a disease that can cause what doctors call a respiratory infection. It can affect the upper respiratory tract (sinus, nose, and throat) or the lower respiratory tract (trachea and lungs). It is caused by a coronavirus called SARS-CoV-2. In December 2019, the first cases were observed in Wuhan, China before developing worldwide [1, 2, 3]. On March 11, 2020, the epidemic was officially confirmed as a pandemic [4].

COVID-19 disease has become the most urgent problem that has attracted worldwide attention quickly due to the rapid increase in the number of new cases.

This new virus is temporarily designated as the new coronavirus 2019 (2019-nCoV) by the World Health Organization (WHO) on January 12, 2020. WHO officially named the disease caused by a coronavirus 2019-nCoV (COVID-19) on February 11, 2020 [5].

COVID-19 has spread to 185 territories, on six continents, and the number of confirmed COVID-19 cases worldwide has exceeded 1.9 million, according to an online virus detection system hosted by Johns Hopkins University on April 14, 2020 [6, 7]. There were 5 countries with 100,000 confirmed positive cases, 17 countries with 1 in 10,000 cases and 50 countries between 1 and 10000 confirmed cases [6]. In Tunisia, there were 200 662 confirmed cases of COVID-19 infection in Tunisia [8] on January 27, 2021.

COVID-19 disease changed the world landscape in just four months. An increased risk of complications has resulted in border closings, rampant storage, empty streets, mass self-

isolation policies. The world has entered a new world war guided by an invisible enemy.

Strategies to fight the virus before April have mainly revolved around containment and tracing - to find and isolate possible cases before they develop. However, with the exponential increase in the number of positive cases, hospitals around the world are quickly overwhelmed and government policies have moved to mitigation, if not acceptance. The frustrating process is further compounded by a lack of testing capacity in the affected countries, which means that the number of positive cases lags behind the number of actual cases. This has led health authorities worldwide to temporarily revert to symptom-based diagnosis and chest x-rays / CT scans based on Artificial Intelligence (AI), which have recently been widely explored for the use of pneumonia detection [9]. These approaches have also been proposed during this pandemic. Their intrinsic experimental nature requires that confirmatory approaches are also used in tandem, as part of a multidisciplinary approach to diagnosis.

Machine learning techniques have been a very important breakthrough in the field of AI in the past few decades. These techniques are widely used to extract tiny features in the analysis of medical images.

In this article, we will explore the feasibility and difficulties of building a diagnostic support system capable of detecting COVID-19 disease in chest radiographs, using a machine learning process and a constrained dataset.

2. Methodology

2.1. Features extraction based on DTM

In recent years, mathematical moments were widely used in several computer vision areas of research such as objects and shape descriptors, pattern recognition, template matching, and image analysis. These moments are scalar quantities used to characterize an image, reflecting significant attributes of it. They provide the most salient characteristics and precise features of representation capability for any given image. The literature distinguishes two kinds of orthogonal moments: the continuous and the discrete. However, discrete moments provide better results than continuous [11].

Several types of mathematical discrete moments have been developed and used as objects feature extraction since they have less redundancy and represent the object more faithfully than the continuous moments such as Zernike or Legendre.

In the present study, we focus on a certain category of discrete mathematical moments called the ‘Discrete Tchebychef Moments’ (DTMs). these moments are obtained by a Discrete Tchebychef Transform (DTT) and they offer very interesting results in terms of object recognition rate of accuracy for object classification [12].

DTM are determined by projecting the input image on to a set of Tchebichef polynomials as follows:

For a given image $f(x, y)$ of size $N \times N$ and value x within range $[0, N - 1]$, a set of tchebichef polynomials $t_n(x)$; $n = 0, 1, \dots, N - 1$, is defined through the use of the following recurrence relation:

$$t_n(x) = \frac{(2n - 1)t_1(x)t_{n-1}(x) - (n - 1)(1 - \frac{(n-1)^2}{N^2})t_{n-2}(x)}{n} \quad (1)$$

Where $t_0(x) = 1$ and $t_1(x) = \frac{2x+1-N}{N}$

Set $\{t_i\}$ has a squared norm as follows:

$$\rho(n, N) = \sum \{t_i(x)\}^2 \quad (2)$$

The Tchebichef moments $T_{mn}(m, n = 0, 1, \dots, S - 1)$ can be represented by:

$$T_{mn} = \frac{1}{\rho(m,S)\rho(n,S)} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} t_m(x)t_n(y)f(x, y) \quad (3)$$

Along spatial coordinates, the repetitive variation patterns are represented by the image texture. Because of such a repetitive property, we can capture the patterns by correlating an image using a set of Tchebichef polynomials.

As a result, a larger magnitude of T_{mn} will lead to a higher correlation. The $F(k)(k = 0, 1, \dots, 2N - 2)$ feature vector can

measure such a correlation while considering the image directions expressed by: $F(k) = \sum_{m+n=k} |T_{mn}|$.

Feature $F(k)$ gives data about the texture properties which is able to be viewed like a texture signature.

2.2. KPCA for dimensionality reduction

The most common technique which is widely used for dimensionality reduction of features is the principal component analysis (PCA) [11]. Despite the proven performance of the PCA technique, it is based on the idea that the system is linear. To address the above issue, a nonlinear PCA method called kernel PCA (KPCA) has been developed in the literature [12]. KPCA can be presented in two steps: the first step is to project the input data onto the feature space through a nonlinear mapping function, and the second step is to implement PCA in that feature space.

KPCA has been widely used to model various nonlinear processes [13,14,15]. In this work, we used KPCA method that reduced the size of the image as following:

Let us consider $X = [x_1, x_2, \dots, x_n]^T$ the data matrix upgrade to unit variance and zero mean.

The mapping of sample in the feature space can be written as:

$$\Phi: E \subset \mathbb{R}^m \rightarrow F \subset \mathbb{R}^H \quad ; \quad x \rightarrow \Phi(x) \quad (4)$$

Where $x \in E \subset \mathbb{R}^m$ is a data vector.

We define the covariance matrix in F as:

$$C_\Phi = \frac{1}{N} \sum_{i=1}^N \Phi_i \Phi_i^T = \frac{1}{n-1} X^T X \quad (5)$$

Where $\Phi = \Phi(x_i) \in \mathbb{R}^N$ and $X = [\Phi(x_1), \Phi(x_2), \dots, \Phi(x_N)]^T \in \mathbb{R}^{N \times h}$. The KPCA reference model (The principal components of data after mapping $\Phi(x_1), \Phi(x_2), \dots, \Phi(x_N)$) is computed by solving the eigenvalue decomposition of the covariance matrix C_Φ in the feature space H such that:

$$\lambda_j \mu_j = C_\Phi \mu_j \quad \text{with } j = 1, \dots, h \quad (6)$$

Where μ_j is the j^{th} eigenvector of C_Φ corresponding to eigenvalue λ_j .

For $\lambda_j \neq 0$, there exist coefficients $\alpha_{i,j}$ $i = 1, \dots, N$ such all eigenvectors μ_j can be considered a linear combination of $[\Phi(x_1), \Phi(x_2), \dots, \Phi(x_N)]$ and can be expressed by:

$$\mu_j = \sum_{i=1}^N \alpha_{i,j} \Phi(x_i) \quad j = 1, \dots, n \quad (7)$$

However, practically, the mapping function Φ is not defined and then C_Φ in the feature space cannot be calculated implicitly. Thus, instead of solving eigenvalue problem

directly on C_ϕ , we apply the kernel trick. The inner product given in (6) may be determined by a kernel function $K(\cdot, \cdot)$ that satisfies Mercer's theorem [16] as follows:

$$\langle \phi(x), \phi(x') \rangle_H = K(x, x') \quad \forall x, x' \in \mathbb{R}^m \quad (8)$$

The definition of a Kernel matrix $K \in \mathbb{R}^{N \times N}$ associated to a kernel function k is the following:

$$K = \begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_N) \\ \vdots & \ddots & \vdots \\ K(x_N, x_1) & \cdots & K(x_N, x_N) \end{bmatrix} \in \mathbb{R}^{N \times N} \quad (9)$$

We can reduce the problem of eigenvalue decomposition of C_ϕ by the application of the kernel matrix. Hence, eigen decomposition of the kernel matrix K is equivalent to performing PCA in \mathbb{R}^H , so that:

$$NAY = KY \quad (10)$$

With:

$\Lambda = \text{diag}(\lambda_1, \dots, \lambda_j, \dots, \lambda_N)$ is the diagonal matrix of eigenvalues λ_j arranged in descending order and

$Y = [\alpha_1, \dots, \alpha_j, \dots, \alpha_N]$ is the matrix of their corresponding eigenvectors.

Since the principal components are orthonormal, it is required to guarantee the normality of μ_j in (7), such that:

$$\langle \mu_j, \mu_j \rangle_H; \quad j = 1 \dots n \quad (11)$$

With N as the number of the first non-zero eigenvalues.

$$\begin{aligned} \langle \mu_j, \mu_j \rangle_H &= \sum_{i,k}^N \alpha_{i,j} \alpha_{k,j} \langle \phi(x_i), \phi(x_k) \rangle_H \\ &= \sum_{i,k}^N \alpha_{i,j} \alpha_{k,j} K_{i,k} = \langle \alpha_j, K \alpha_j \rangle_H \\ &= \langle \alpha_j, K \alpha_j \rangle_H = \lambda_j \langle \alpha_j, \alpha_j \rangle_H \end{aligned} \quad (12)$$

Where $K_{i,k} = K(x_i, x_k)$. The corresponding eigenvectors α_j must be scaled as:

$$\langle \alpha_j, \alpha_j \rangle_H = \|\alpha_j\|^2 = \frac{1}{\lambda_j}; \quad j = 1 \dots n \quad (13)$$

Many kernel functions have been proposed in literature such as: Linear Kernel, Polynomial kernel, Radial basis function kernel (RBF).

The Radial Basis Function (RBF) is studied such that:

$$K(x, y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right) \quad (14)$$

where $\sigma \in \mathbb{R}^+$ Mean centering the training data in the feature space should be Performed $\sum_{i=1}^n \phi_i = 0$ if this is not the case, kernel matrix K should be scaled such that :

$$\bar{K} = FKF \quad \text{With } F = (I_n - E) \quad (15)$$

Where $1_n = \frac{1}{n} [1, \dots, 1]^T \in \mathbb{R}^n$ and E is an $n \times n$ matrix with elements $\frac{1}{n}$.

3. Kernel-ELM Algorithm

The ELM has been considered as one of the Single Layer Feedforward Networks (SLFN) put forward in [26]. The ELM was suggested for training the SLFN in the first place and then extended to train the SLFN in a general way, where hidden layers did not need to be neurons.

The used structure in the ELM is typically an SLFN one that comprises three layers: input, hidden, and output (Fig. 1). Through the use of a weighted connection named *weight* (w), each neuron is linked. In addition to that, other parameters are utilized; we can mention mainly the *bias* (b) which provides further adaptable model parameters, and the activation function (g) that computes output (y). It is possible for transfer function f to be sigmoid-functional, radial-based, Tang-hyperbolic, linear, or logarithmic. Thus, the network could be described utilizing triplet (w, b, g). Fig.3 illustrates the ELM structure.

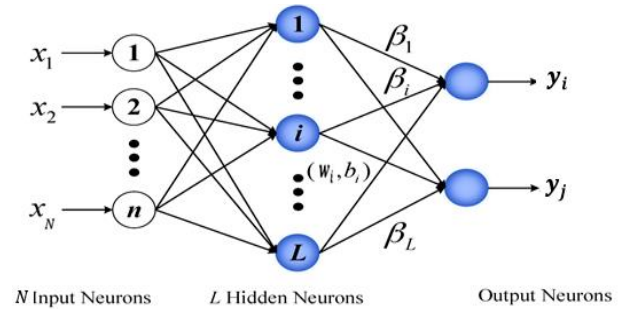


Fig.1.ELM structure

Let consider the training samples $X = \{x_i, y_i\}_{i=1, \dots, N}$ with $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in \mathbb{R}^n$, $y_i = [y_{i1}, y_{i2}, \dots, y_{im}]^T \in \mathbb{R}^m$ where m, d , and N represent, in the order given, the number of output nodes, the dimension, and the number of instances. On the one hand, $m = 1$ for regression problems. On the other hand, m stands for the number of labels, classes or categories for classification problems.

The ELM output function of the SLFN with L hidden layer nodes is expressed as:

$$f(x_i) = y_i = \sum_{j=1}^k \beta_j h(w_j \cdot x_i + b_j) \quad i = 1, \dots, N \quad (16)$$

where $w_j = [w_{j1}, w_{j2}, \dots, w_{jn}]$ is the weight vector that links up between input neurons and the j^{th} hidden neuron, $\beta_j = [\beta_{j1}, \beta_{j2}, \dots, \beta_{jm}]^T$ is the weight vector that ties in an output neuron with the j^{th} hidden neuron, h is a sigmoid function, $w_j \cdot x_i$ indicates an inner product of w_j and x_i , b_j is the bias of the j^{th} hidden neuron, and k is the hidden neurons number in a hidden layer.

We can write equation (17) in a matrix format as: $H\beta = Y$. In this case, H represents a neural network hidden-layer output matrix:

$$H = \begin{bmatrix} h(w_1 \cdot x_1 + b_1) & \cdots & h(w_k \cdot x_1 + b_k) \\ \vdots & \ddots & \vdots \\ h(w_1 \cdot x_N + b_1) & \cdots & h(w_k \cdot x_N + b_k) \end{bmatrix}_{N \times k} \quad (17)$$

$$\text{With } \beta = \begin{pmatrix} \beta_1^T \\ \vdots \\ \beta_k^T \end{pmatrix}_{k \times m} \quad \text{and} \quad Y = \begin{pmatrix} y_1^T \\ \vdots \\ y_k^T \end{pmatrix}_{N \times m}$$

In the condition when the hidden layer bias and the input weight are generated in a random way according to the H output matrix, then we consider that SFLN learning is merely and essentially equal to finding a least square solution. In fact, this is mathematically modeled as: $\beta = H^+Y$

In such a case, H^+ is Moore-Penrose pseudo-inverse [27] of H in a way that: $H^+ = \left(\frac{I}{C} + H^T \cdot H\right)^{-1} \cdot H^T$

Hence, the ELM output is:

$$f(x) = h(x)\beta = h(x) \left(\frac{I}{C} + H^T H\right)^{-1} H^T Y \quad (18)$$

For any attribute x , $label(x) = sign(h(x)\beta)$

As for the ELM with kernels, in general, very good regression and classification accuracy can be obtained through kernel introduction. According to [28], when $h(x)$ is unknown (that is to say the function is implicit), Mercer's conditions can be applied up on the ELM and an ELM kernel matrix is defined as follows:

$$K_{ELM} = HH^T: K_{ELM}(i, j) = h(x_i) \cdot h(x_j) = K(x_i, x_j) \quad (19)$$

In the KELM, $H = [h(x_1)^T \dots h(x_N)^T]^T$ is a hidden layer output matrix whose role is to map the x_i data into a hidden layer feature space from an input space. This is beside the point for the target value y_i as well as for the number of output nodes m . As illustrated in Table 2, the $K_{ELM} = HH^T$ kernel matrix is just linked with the x_i input data as well as the number of training samples N . This is in particular for multi class classification, binary classification and regression. Then, it is possible to write the ELM classifier output function (19) in a compact way as:

$$f(x) = \left(\begin{bmatrix} K(x, x_1) \\ \vdots \\ K(x, x_N) \end{bmatrix}^T \left(\frac{I}{C} + K_{ELM} \right)^{-1} Y \right) \quad (20)$$

TABLE 1. THREE COMMON KERNELS WITH THEIR PARAMETERS AND FORMULA

Name	Formula and parameters
RBF kernel (RBF_KELM)	$K_\sigma(x, y) = \exp\left(-\frac{1}{2} \frac{\ x - y\ ^2}{\sigma^2}\right)$ σ is parameter of Rbf kernel
Polynomial kernel (Poly-KELM)	$K_p(x, y) = (x \cdot y + 1)^p$ p is the degree of the polynomial
Linear kernel (Lin-KELM)	$K_l(x, y) = x \cdot y$

4. Experimental Evaluation

4.1. Datasets used

To target the issue at hand, we've collected own dataset, combining the Kaggle Chest X-ray dataset [18] with the COVID19 Chest X-ray dataset [19] collected by Dr. Joseph Paul Cohen of the University of Montreal. Both of these datasets consist of posterior anterior chest images of patients with pneumonia. As the COVID19 dataset is being updated daily as more cases are published

A. Statistical analysis

In order to show the effectiveness of our approach, objective evaluation criteria are measured: sensitivity, specificity, accuracy and F-score. They are described below. The definition of TPR, FNR, FPR and TNR are illustrated in Table 1.

TABLE 2. DEFINITION OF TPR, FNR, FPR AND TNR

	Total population	Predicted class	
		Pneumonia (covid19)	Normal
True class	Pneumonia (covid19)	TPR	FNR
	Normal	FPR	TNR

Where TPR and TNR respectively mean True Positive Rate and True Negative Rate, which determine the number of pneumonia (COVID-19) cases classified as such (TPR) and the number of normal cases classified as such (TNR). FPR and FNR respectively denote False Positive Rate and False Negative Rate, which represent the classification of pneumonia (COVID-19) cases in normal cases (FNR) and normal cases in pneumonia (COVID-19) cases (FPR). The specificity (Sp) of a classifier measures the ability to correctly detect normal cases, while the definition of the sensitivity (Se) is given as the TPR instances and FNR instances that have been classified as TPR. This measure is used in the medical field as it gives knowledge about the number of cases that are correctly identified either as malignant (COVID-19 cases) or as benign (normal cases). It is the ability of the model to find all the relevant cases in the dataset.

$$Sensitivity = \frac{TPR}{TPR + FNR} \quad (21)$$

$$Specificity = \frac{TNR}{TNR + FPR} \quad (22)$$

$$Accuracy = \frac{TPR + TNR}{TPR + FPR + TNR + FNR} \quad (23)$$

$$F_{score} = 2 \times \frac{TPR}{P} \times \frac{TPR}{Q} \div \left(\frac{TPR}{P} + \frac{TPR}{Q} \right) \quad (24)$$

With: $P = TPR + FPR$ and $Q = TPR + FNR$

4.2. Results and discussion

The proposed methodology has experimented with our own dataset collected by combining the Kaggle Chest X-ray dataset with the COVID19 Chest X-ray dataset collected by Dr. Joseph Paul Cohen of the Université de Montréal. These two datasets consist of posterior and anterior chest images of patients with pneumonia. The images collected are classified as normal or pathological using the proposed methodology. The performance of the proposed model is evaluated according to different performance measures, namely sensitivity, specificity, and success rate (ACC). Before the feature extraction module, images are preprocessed with CLAHE to improve contrast.

Next, the DTT technique is applied to the images to obtain the characteristic matrix. Thus, in order to reduce the size of the vector of characteristics and to simplify the classification, KPCA is used, which reduces the number of characteristics by preserving 95% of the variance of the original data. Reduced functionality is passed to the three Poly-KELM, Lin-KELM, and Rbf-KELM algorithms to classify images as normal or pathological.

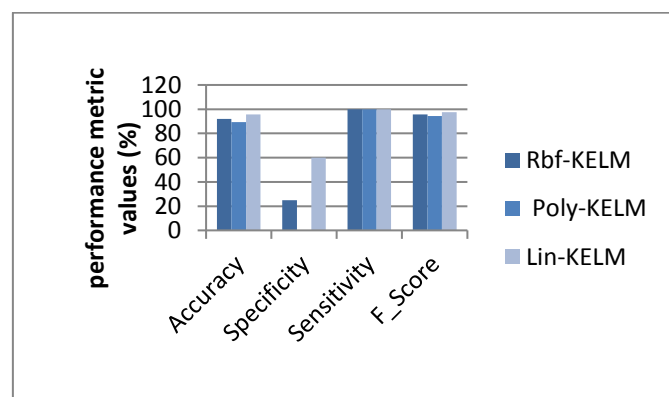


Fig 2 Bar chart of classification results

Figure 2 shows the results of accuracy, specificity, and F_score of different classifiers obtained.

Based on the figure above, it was verified that when using RBF-KELM and Lin-KELM we obtained better results, which were 92% and 95.6% of accuracy, 25% and 60% of specificity, 100% and 99.8% of sensitivity, and a F_score of 95.71% and 97.59%, respectively. On the other hand, Poly-KELM classifiers showed to be more efficient.

TABLE 3 COMPARISON OF CONFUSION MATRIX OF THE THREE CLASSIFICATION METHODS

Classifier	TPR	FPR	TNR	FNR
Rbf-KELM	670	60	20	0
Poly-KELM	670	80	0	0
Lin-KELM	669	32	48	1

In order to demonstrate the superior performance of different classifiers, a detailed comparison in terms of TPR, TNR, FPR, and FNR is shown in Table 2.

TABLE 4. COMPUTATIONAL COST OF THE THREE CLASSIFICATION METHODS

Classifier	Train-Time (s)	Test-Time (s)
Rbf-KELM	0.015	0.013
Poly-KELM	0.11	0.05
Lin-KELM	0.007	0.002

From the above results, we can conclude that the RBF-KELM and the Lin-KELM gives higher classification performance than the Poly-KELM method with the best training and testing times.

5. Conclusion and feature works

In this work, we present a method of classifying the textural characteristics of chest x-ray images using the RBF-KELM, Poly-KELM, and Lin-KELM classifiers to diagnose COVID-19. To target the problem, we collected our own dataset, combining the Kaggle chest x-ray dataset with the COVID19 chest x-ray dataset. The RBF-KELM and Lin-KELM methods performed better than the Poly-KELM algorithm. From the above results, we can conclude that the introduction of the RBF and linear kernel function for KELM as classification techniques gives greater precision..

For our next research perspectives, we will then have to implement the prototype of the method in future portable devices that we will develop to automatically classify COVID 19 disease using radiographic images of the chest.

References

- [1] Zhu, N., Zhang, D., Wang, W., Li, X., Yang, B., Song, J., ... & Niu, P. (2020). A novel coronavirus from patients with pneumonia in China, 2019. *New England Journal of Medicine*.
- [2] Perlman, S. (2020). Another decade, another coronavirus.
- [3] Hui, D. S., Azhar, E. I., Madani, T. A., Ntoumi, F., Kock, R., Dar, O., ... & Zumla, A. (2020). The continuing 2019-nCoV epidemic threat of novel coronaviruses to global health—The latest 2019 novel coronavirus outbreak in Wuhan, China. *International Journal of Infectious Diseases*, 91, 264.

- [4] "WHO Director-General's opening remarks at the media briefing on COVID-19 - 11 March 2020". Who.int. 2020.
- [5] WHO 2020, "We now have a name for the #2019nCoV disease: COVID-19. I'll spell it: C-O-V-I-D hyphen one nine – COVID-19", Tweet, 11 February, viewed 11 February 2020, <https://twitter.com/WHO/status/1227248333871173632>
- [6] Wuhan Coronavirus (2019-nCoV) Global Cases (by Johns Hopkins CSSE). Case Dashboard. [accessed 15 April 2020].
- [7] Wang, D., Hu, B., Hu, C., Zhu, F., Liu, X., Zhang, J., ... & Zhao, Y. (2020). Clinical characteristics of 138 hospitalized patients with 2019 novel coronavirus–infected pneumonia in Wuhan, China. *Jama*, 323(11), 1061-1069.
- [8] www.santetunisie.rns.tn
- [9] Chumbita, M., Cillóniz, C., Puerta-Alcalde, P., Moreno-García, E., Sanjuan, G., Garcia-Pouton, N., ... & Garcia-Vidal, C. (2020). Can Artificial Intelligence Improve the Management of Pneumonia. *Journal of Clinical Medicine*, 9(1), 248.
- [10] Marcos JV, Cristóbal G (2013) Texture classification using discrete Tchebichef moments. *JOSA A* 30(8):1580
- [11] Wang, Q. (2012). Kernel principal component analysis and its applications in face recognition and active shape models. arXiv preprint arXiv:1207.3538.
- [12] Schölkopf, B., Smola, A., & Müller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5), 1299-1319.
- [13] Jaffel I, Taouali O, Harkat MF, Messaoud H (2018) Fault detection and isolation in nonlinear systems with partial Reduced Kernel Principal Component Analysis method. *Trans Inst Meas Control* 40(4):1289–1296.
- [14] Fazai R, Taouali O, Harkat MF, Bouguila N (2016) A new fault detection method for nonlinear process monitoring. *Int J Adv Manuf Technol* 87(9-12):3425–3436.
- [15] Taouali O, Jaffel I, Lahdhiri H, Harkat MF, Messaoud H (2016) New fault detection method based on reduced kernel principal component analysis (RKPCA). *Int J Adv Manuf Technol* 85(5-8):1547–1552
- [16] Mercer J (1909) Functions of positive and negative type and their connection with the theory of integral equations. *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character* 209:415–446.
- [17] Storn, R. and Price, K. (1997). Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4), 341–359.
- [18] <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>
- [19] <https://github.com/ieee8023/covid-chestxray-dataset>

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0
https://creativecommons.org/licenses/by/4.0/deed.en_US