# Fuzzy Logic modeling for Objective Image Quality Assessment

Ghislain Takam Tchendjou, Rshdee Alhakim, Emmanuel Simeu
Université Grenoble Alpes, TIMA, F-38000, Grenoble, France
CNRS, TIMA, F-38000, Grenoble, France
Ghislain.Takam@imag.fr, Rshdee.Alhakim@imag.fr, Emmanuel.Simeu@imag.fr

***Abstract*—This paper presents a novel methodology of objective image quality assessment (IQA) based on Fuzzy Logic (FL) method. The main purpose is to automatically assess the quality of image in agreement with human visual perception. The used attributes (quality metrics) and evaluation criteria (human rating mean opinion score MOS) are extracted from image quality database TID2013. The fuzzy model design starts by selecting the most independent attributes, by applying Pearson's correlation approach and seeking the most correlated metrics with the corresponding MOS. Then, Adaptive Neuro-Fuzzy Inference System (ANFIS) is applied in order to construct an objective fuzzy model able to efficiently predict the image quality correlated with the subjective MOS. In this paper, different fuzzy models are produced by modifying certain ANFIS configurations. After that, we select the appropriate ANFIS model that provides high prediction accuracy and stability with taking into account its implementation complexity. The overall architecture of the selected FL model consists of four input metrics, two bell-shaped membership functions associated to each input metric, two fuzzy if-then rules, two linear combination equations and one output which gives the image adequate quality score. Finally the performance of the proposed fuzzy model is compared with other IQA models produced by different machine learning methods, the simulation results demonstrate that the fuzzy logic model has a high stable behavior with the best agreement with human visual perception.**

**Keywords - Image Quality Assessment; Machine Learning; Fuzzy Logic; Fuzzy Inference System, ANFIS.**

## I. Introduction

The judgment of the image quality has become a hot topic in image processing research since it plays a significant role in enhancing visual artifact concealment process and in evaluating the performance of many visual processing applications in several domain such as medical service, security service, entertainment sector and automated traffic control [1], [2].

The digital visual processing of an image usually consists of five steps: 1) the acquisition, 2) the compression, 3) the transmission, 4) the storage and 5) the display [3]. Each step induces more or less significant distortions to the initial image and degrades the image quality [4].

The image quality assessment methods are classified into two main categories: subjective and objective. In subjective methods, image quality is evaluated based on a group of human observers who watches images and rates their quality; the quality scores will then be gathered from all observers and statistically processed in order to obtain the Mean Opinion Score (MOS) [5]. While this method can offer accurate scoring, it is inherently noisy, expensive and very time-consuming to acquire [6], these drawbacks make this method impractical for real-time applications.

In objective methods, the goal is to provide a computational models that automatically assess image quality, as illustrated in Fig. 1. Several recent rechearchs have exploited machine learning (ML) algorithms (such as artificial neural network (ANN), support vector machine (SVM), non-linear regression (NLR), decision tree (DT) etc.) to realize an objective IQA model taking into account the agreement with human perception [4], [5], [7], [8]. Despite subjective IQA methods are not appropriate for real time application, they are the best way to evaluate and benchmark objective IQA algorithms.

To measure the correlation between the objective and subjective quality scores, four coefficients are used: 1) Pearson's linear correlation coefficient (PLCC) which is used to measure the degree of the relationship between linear related variables; 2) Spearman's rank order correlation coefficient (SROCC) which is used to measure the prediction monotonicity and the degree of association between two variables; 3) Kendall's rank order correlation coefficient (KROCC) which is used to measure the strength of dependence between two variables; and 4) coefficient of determination ($R^2$) which indicates the proportion of the variance in the dependent variable (MOS) that is predictable from the independent variables (Metrics). In addition to these four correlation coefficients, mean squared error (MSE) is also used to measure the difference between the subjective MOS and its estimated value [1], [9].

The purpose of this paper is to design a novel objective model for Image Quality Assessment based on Fuzzy Logic (FL) algorithms, then the derived FL model will be evaluated and compared with the IQA models produced by other machine learning methods, mentioned in section III.

The paper is organized as follows: Section II gives an overview of the image quality database used; and explains briefly how to select the most significant image quality metrics in this database. Section III mentions five different ML algorithms used in previous works for building IQA model. Section IV explains the architecture and configurations of Fuzzy Logic systems, then ANFIS is applied to produce an appropriate fuzzy model able to efficiently predict image quality. The simulation results in Section V compare the prediction performance of ANFIS model to the five other ML

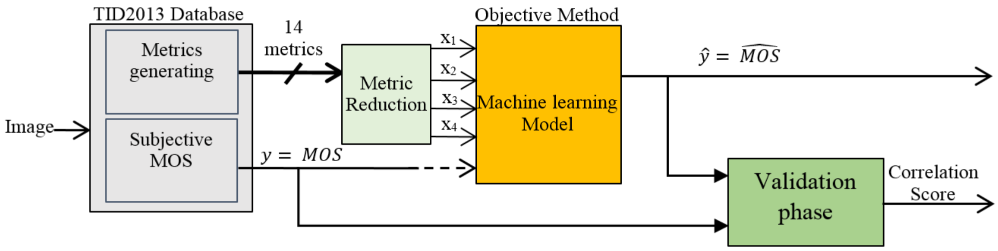



Fig. 1. Objective IQA Process.

Fig. 2. Reference images of TID2013.

models. Finally Section VI draws a conclusion.

## II. Image Quality Database

Several available publicly databases could be used to evaluate and benchmark the objectives IQA algorithms (such as TID2008, TID2013, LIVE Image, CSIQ, IVC-LAR, etc.) [5]. The work in this paper is evaluated by Tampere Image Database 2013 (TID2013) [10], [11]. TID2013 has 3000 distorted images extracted from 25 reference images (shown in Fig. 2), 24 types of distortions and 5 different levels for each distortion. Each distorted image in TID2013 is related to an adequate MOS value which has been derived from the results of 985 experiments carried out by observers from five countries (Finland, France, Italy, Ukraine, and USA), all MOSs range between 0 and 9 [10], [11]. In addition to MOS, TID2013 also provides 14 image quality metrics to characterize each distorted image (such as Peak Signal to Noise Ratio PSNR, Feature SIMilarity FSIM, Structural SIMilarity SSIM, etc.).

Although we could use all 14 provided quality metrics to design a relatively good MOS estimator. However that would lead to increase model-structure complexity and training-time consumption. Moreover, these metrics could be highly correlated and share redundant information, causing a decrease in the prediction accuracy (overfitting problem). By using Pearson's correlation we select the most pertinent metrics which have simultaneously high correlation with MOS and low cross-correlation [12]. As a result, the number of metrics is reduced to four, and the reduced-size TID2013 is composed of N = 3000 image datasets on the form $(X_1, Y_1), ..., (X_N, Y_N)$ where $X_i$ is the image quality metric vector: $X_i = (x_{i1}\ x_{i2}\ x_{i3}\ x_{i4})^T$ corresponding to FSIMc, PSNR-HMA, PSNRc and SSIM respectively, and $Y_i$ is the corresponding subjective MOS, where i represents the index of image set [12].

## III. Machine Learning Methods

As described in Fig. 1, after reducing the number of inputs into four significant metrics, the next step is to construct a suitable ML model for automatically predicting the perceived quality of any input image. The construction process of objective IQA block essentially consists of training and validation phases: During the training phase, ML algorithms are applied to create an appropriate mapping model that relates the quality metric vector of image to the provided subjective MOS of the same image. While the validation phase is used to estimate how well the trained ML models predict the perceived image quality, and to compare the accuracy and robustness of different ML methods. In this phase, we apply Monte Carlo Cross-Validation (MCCV) method (termed also Random Sub-sampling Cross-Validation) [13], as follows: a) First, we split the data-set with a fixed fraction $(\alpha\%)$ into training and validation set; where $\alpha\%$ of the samples are randomly assigned to the training process in order to design a model, while the remaining samples are used for assessing the predictive accuracy of the model. b) Then, the first step is repeated k times. c) Finally, the correlation scores are averaged over k. In this paper, $\alpha\% = 70\%$ and k = 1000.

Before applying fuzzy logic algorithms to create IQA model, let us first present briefly the other five ML methods used in previous research works [5], [7], [8], [12], which are: linear discriminant analysis, k-nearest neighbor, artificial neural network, nonlinear regression and decision tree. In simulation results section we will see the performance comparison of IQA model produced by using all these ML methods.

### A. Linear Discriminant Analysis (LDA)

Linear discriminant analysis is usually used in statistic and machine learning, to find a linear combination of features that separates or classifies objects (such as images, products, customers, etc.) into two or more groups [12], [14].

### B. k-Nearest Neighbors (k-NN)

The key concept in the k-nearest neighbors is to project all training set points on the metric space, then calculate the distance from these observation points and the point which we want to classify; and select the k-closest points. Finally the class corresponding to the studied point is determined by applying majority vote rule [12], [15].

### C. Artificial Neural Network (ANN)

Artificial neural network is a sophisticated learning method, inspired from the structure of biological neural networks, able to model nonlinear relationships between inputs and outputs. In [5], [7] the structure of ANN model used consists of two layers: hidden and output layers; the hidden layer is composed of four interconnected nodes and the logistic sigmoid activation function, while the output layer contains one node and a linear activation function. Standard Matlab tools were employed for ANN training and selecting the network coefficients.

### D. Non-Linear Regression (NLR)

The polynomial non-linear regression method seeks to express MOS as an appropriate nonlinear polynomial combination of the metric variable, as follows:

$$\widehat{y} = a_0 + \sum a_i x_j + \sum b_i x_j x_k + \sum c_i x_j x_k x_l + ... \quad (1)$$

where $a_0$ is a constant, $a_i$, $b_i$, $c_i$ are respectively the first, second and third term coefficients, the equation indexes $j, kandl \in \{1, 2, 3, 4\}$. The papers [5], [8] are explained how to design NLR model (select the terms and coefficients of polynomial), and shown that the image quality can efficiently be predicted with a polynomial regression model of degree 3 with 19 terms [5].

### E. Decision Tree (DT)

Decision tree is a non-parametric and nonlinear method, built through a recursive binary-partitioning process. In [5], [16], they applied DT Matlab tools, in order to seek the mapping between the quality metrics and MOSs, and consequently the generated mapping model from the training phase has a binary tree structure with 39 levels in which each internal node represents a test on one input metric, each tree branch represents the outcome of the test and each terminal leaf node contains the predicted MOS value.

## IV. Fuzzy Logic

The fuzzy logic method can be considered as a generalization of the classical set theory by introducing the notion of degree in the verification of a condition; thus enabling a condition to be in a state other than 0 or 1. It provides a flexibility for reasoning, which makes it possible to take into account inaccuracies and uncertainties [17]. The general architecture of a fuzzy logic system, illustrated in Fig. 3, can be described as follows:

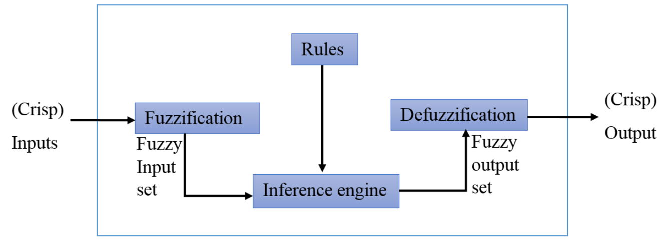



Fig. 3. General architecture of a fuzzy logic system.

- Fuzzification module: in this stage, the system input is transformed from its crisp number into fuzzy sets.
- Knowledge base (Rules): which is represented by a set of IF-THEN rules provides by experts.
- Inferences engine: which simulates the process of human reasoning. This is done by making fuzzy inference on the inputs using the rules.
- Defuzzification module: it transforms the fuzzy set obtained by the inference engine into a crisp value.

Since IQA is basically a nonlinear problem, a solution using fuzzy logic technique could be suitable for it. In our study case, the fuzzy inference system (FIS) has four input metrics $X = \{x_1, x_2, x_3, x_4\}$ (previously mentioned) and one output $y = MOS$. The proposed FIS model is, a Fuzzy Sugeno model [17], composed of if-then rules with the following form:

$$\begin{aligned} &Rule\ i: \\ &if \quad x_1\ in\ A_{i,1},\ x_2\ in\ A_{i,2},\ x_3\ in\ A_{i,3}\ and\ x_4\ in\ A_{i,4} \\ &\qquad then\ f_i = \ a_{i0} + a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 + a_{i4}x_4 \end{aligned} \quad (2)$$

where $A(i,j)$ represents the linguistic label of the input metric $j$ in the rule $i$, so the index $i$ refers to the rule number, while the index $j \in \{1, 2, 3, 4\}$ refers to the input metric number, $f_i$ is the output of the rule $i$ and it is a linear combination of the input variables with linear design parameters $\{a_{i0}, a_{i1}, a_{i2}, a_{i3}, a_{i4}\}$ that are determined during the training phase.

The fuzzy logic model can automatically be generated by applying Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS is a combination of artificial neural network (ANN) and fuzzy inference systems; it works by extracting a training set of input-output data vector from TID2013, and applying the intelligent learning technics of the ANN in order to derive the suitable fuzzy if-then rule set that permits minimizing the error between subjective MOS and the estimated MOS given by FL model. A typical ANFIS structure, as shown in Fig. 4, is composed of five layers [17], [18] :

- Layer 1: the first layer computes the degree of the membership for each input. Every node in this layer is an adaptive node with the following function:

$$O^1_{i,j} = \mu_{A_{i,j}}(x_j), \qquad j \in \{1, 2, 3, 4\} \quad (3)$$

Where $O^1_{i,j}$ denotes the membership function (MF) of $A_{i,j}$ that specifies the degree (between 0 and 1) to which

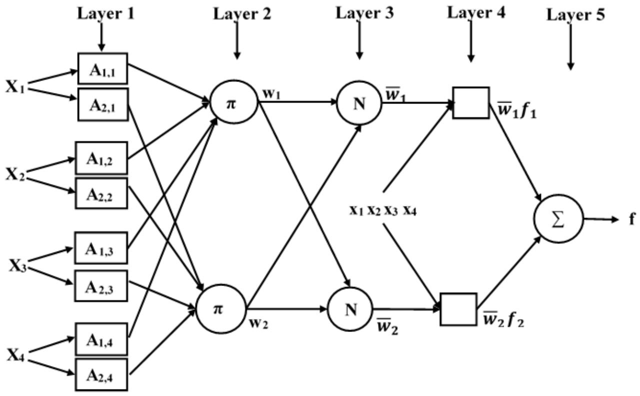



Fig. 4. ANFIS Architecture with two rules.

the given input $x_j$ satisfies$A_{i,j}$. The most commonly MF families used in practice encompass: Gaussian, Triangular, Trapezoidal, Generalized Bell and S-shaped MFs.

- Layer 2: this layer calculates the weight $w_i$ (called also the firing strength) of the $i$th rule by multiplying the incoming signals:

$$O_i^2 = w_i = \prod_{j=1}^{4} \mu_{A_{i,j}}(x_\mathrm{j})$$
$$= \mu_{A_{i,1}}(x_1) \times \mu_{A_{i,2}}(x_2) \times \mu_{A_{i,3}}(x_3) \times \mu_{A_{i,4}}(x_4) \quad (4)$$

- Layer 3: This layer normalizes all the forth firing strengths as follows:

$$O_i^3 = \overline{w_i} = \frac{w_i}{\sum_i w_\mathrm{i}} \quad (5)$$

- Layer 4: the output of this layer is the product of the normalized firing strength and the corresponding rule function output.

$$O_i^4 = \overline{w_i} f_i \quad (6)$$

- Layer 5: it computes the overall output as the addition of all incoming signals.

$$O^5 = \widehat{MOS} = \sum_i \overline{w_i} f_i \quad (7)$$

The fuzzy network nodes in Fig. 4 contains square-shaped nodes in layer-1 and layer-4 and circle-shaped nodes in the rest layers: circle means that the corresponding node is fixed, while square node is an adaptive node with parameters which require to be adjusted in order to achieve a desired mapping between the input metrics and the output MOS. It is worth mentioned that the layer-1 parameters are nonlinear and control the characteristics of membership functions (position, width,...); while the layer-4 parameters are linear and concern the polynomial combination functions.

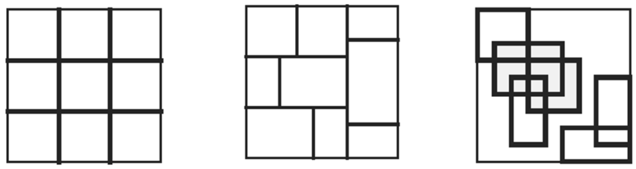
Having briefly presented the general structure of ANFIS network with five layers, the aim of next step is to modify certain configurations of the ANFIS network before launching training phase. These modifications permit to produce several fuzzy ANFIS models to predict the quality of the images. Then we study the prediction performance of the resulting ANFIS models. Finally we select the fuzzy model that achieves high prediction accuracy and good stability, taking into account its circuit complexity. These ANFIS configurations include: partitioning strategy applied to divide the input space, the number of training epochs, the number of membership functions for each input variables and type of these MFs.

### A. Partitioning Strategy

Actually each fuzzy rule is composed of two parts: the first is If-part (antecedent of the fuzzy rule) which covers a particular patch of the input space, while the second part is Then-part (consequent constituent of the fuzzy rule) which describes the behavior within this patch represented mathematically by a linear combination function. Thus, we seek first to partition the metric space into several local patches, and construct later a suitable fuzzy rule for each patch. Fuzzy partitioning technics can be classified into three main categories: grid partitioning, tree partitioning and scatter partitioning [17], [19]. Fig. 5 illustrates the three methods for partitioning two-dimensional input space. The grid partition strategy seeks to divide the data space into very regular sub-spaces, as shown in Fig. 5 (a). Despite the fact that this strategy requires a relatively small number of MFs for each input in order to formulate all fuzzy rules, it has a serious disadvantage that it needs to produce a fuzzy rule set with a significantly big size in order to provide an acceptable FIS performance [20], [21]. For example, if we want to design a fuzzy IQA model with four inputs and three membership functions for each input, this partition strategy would produce $3^4$ = 81 fuzzy if-then rules. That leads to increase both the number of linear rule parameters calculated during the training phase and the circuit complexity of the overall IQA system. In order to resolve this problem and reduce the model complexity, the other partition strategies could be used. The tree partition technology resolves the mentioned problem but it is needs to increase the number of MFs for each input variable. Additionally the resulting fuzzy model is usually less descriptive [19]. In this paper, we apply the third partition strategy based on clustering algorithm in order to reduce the number of rules to a reasonable amount. This strategy is the most flexible way to partition the metric space and it allows also the If-parts of the fuzzy rules cover the arbitrary locations in the metric space [20], [21]. Fuzzy logic toolbox in Matlab provides a built-in function (genfis2), allowing to initialize the input-space partition into clusters based on scatter partition strategy, then the training process of ANFIS optimizes the location and size of these clusters inside the input space. One restriction of genfis2 function is that different rules cannot share the same linguistic label $A_{i,j}$ of the input [18], [22]. To simplify, we assume that all the input variables have the same number of the linguistic labels equal to number of rules.

### B. Number of Training Epochs

ANFIS applies a hybrid learning algorithm on the given training data set to tune the adaptive parameters. The hybrid



(a) *Grid partition* (b) *Tree partition* (c) *Scatter partition*

Fig. 5. Fuzzy partitioning techniques.

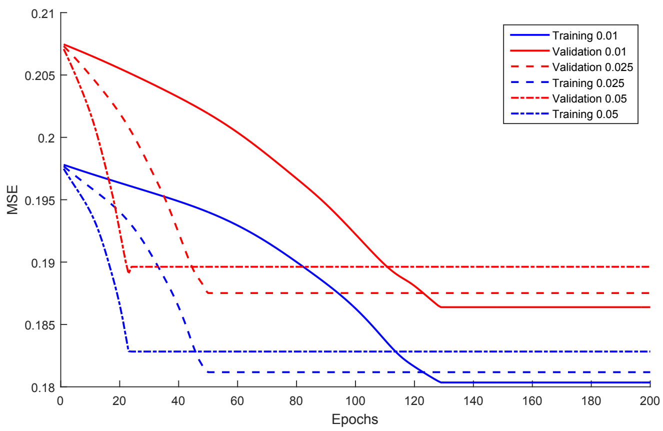



Fig. 6. Training and validation error curves for different epoch sizes.

learning algorithm is considered as a combination of least-squares algorithm (LS) and gradient descent algorithm (GD) [17], [23]. This hybrid learning procedure is performed in several iterations (epochs), each training epoch is composed of forward and backward passes. In the forward pass, the output linear parameters are updated by the LS method, while the input nonlinear parameters are identified by GD method in the backward pass [23]. Fig. 6 shows the evaluation of the fuzzy model performance over the number of epochs and taking into account the initial step size of the gradient descendant algorithm. Fig. 6 demonstrates clearly that a decrease of the initial step size ($\beta$) leads to slightly reduce the final training and checking errors, so to improve the estimation performance, but in the price of the speed of convergence. For example, since $\beta = 0.01$ the estimator takes 130 training epochs to converge to a solution with checking $MSE = 0.1864$. While, since $\beta = 0.05$ the estimator takes only 23 training epochs in order to reach a solution with checking $MSE = 0.1891$. We can also notice that once the optimization algorithm reaches the final solution, the performance remains stable even with increasing the number of training epochs. We would like to mention that the same simulation was run several times for different MF type and for different number of rules and we got the same results of Fig. 6. For the next simulations, the number of epochs and initial step size are set to 23 and 0.05 respectively in order to reduce the training time.

### C. Membership Function Type

Having fixed the partitioning strategy and desired number of epochs, the next step briefly explains the structure of the following MF families: Gaussian, Triangular, Trapezoidal, Generalized Bell and S-shaped MFs (drawn in Fig. 7) [24]. Then we will study the prediction performance and implementation complexity for fuzzy IQA models produced by different type of membership function.

1) The Gaussian membership function is defined as given below, where the function depends on two parameters $\{\sigma, c\}$: $c$ represents the function center and $\sigma$ controls the width.

$$\mu_A(x;\sigma,c) = exp(\frac{-(x-c)^2}{2\sigma^2}) \tag{8}$$

2) The generalized bell MF is calculated as given below, where $\{a, b, c\}$ forms the parameter set. We can adjust $c$ and $a$ to vary the center and width of the function respectively, while the parameter $b$ is usually positive and used to control the slopes.

$$\mu_A(x;a,b,c) = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}} \tag{9}$$

3) The triangular MF is calculated as given below, where this function is defined by three parameters $\{a, b, c\}$: $a$ and $c$ locate the feet of the triangle and $b$ locates the peak.

$$\mu_A(x;a,b,c) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ \frac{c-x}{c-b}, & b \le x \le c \\ 0, & c \le x \end{cases} \tag{10}$$

4) The trapezoidal MF is calculated as given below, where trapezoidal MF is specified by four parameters $\{a, b, c, d\}$ :$a$ and $d$ locate the feet of the trapezoid and $b$ and $c$ locate the shoulder.

$$\mu_A(x;a,b,c,d) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \frac{d-x}{d-c}, & c \le x \le d \\ 0, & d \le x \end{cases} \tag{11}$$

5) The S-shaped MF is calculated as given below, where the function parameters $\{a, b\}$ locate the extremes of the sloped portion of the curve.

$$\mu_A(x;a,b) = \begin{cases} 0, & x \le a \\ 2\left(\frac{x-a}{b-a}\right)^2, & a \le x \le \frac{a+b}{2} \\ 1-2\left(\frac{x-b}{b-a}\right)^2, & \frac{a+b}{2} \le x \le b \\ 1, & b \le x \end{cases} \tag{12}$$

Table I illustrates the number of parameters and execution complexity for MF previously mentioned. For instance, Gaussian MF curve is defined by two adaptive parameters and its architecture consists basically of four arithmetic units: one addition, two multiplications and one exponential. Fig. 8

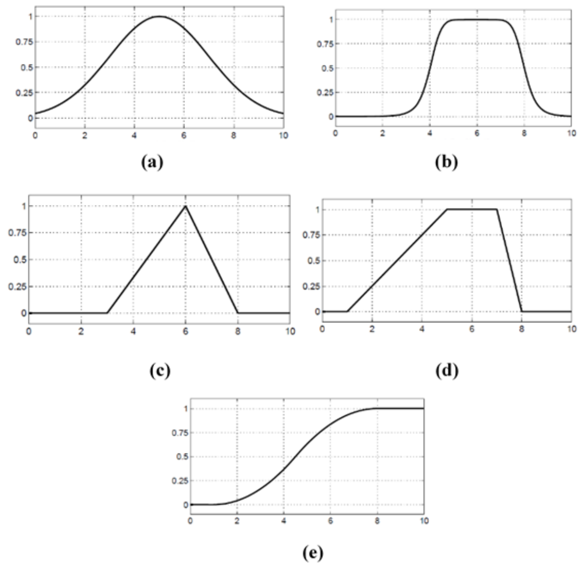



Fig. 7. Example of Membership function. (a) Gaussian MF with $\{\sigma=2,c=5\}$; (b) Bell MF with $\{a=2,b=4,c=6\}$; (c) Triangular MF with $\{a=3,b=6,c=8\}$; (d) Trapezoidal MF with$\{a=1,b=5,c=7,d=8\}$; (e) S-shaped MF with $\{a=1,b=8\}$.

TABLE I
STRUCTURE COMPLEXITY FOR DIFFERENT TYPES OF MF.

| **MF** | **Matlab Function** | **# param.** | **Execution complexity** |
|---|---|---|---|
| Gaussian | Gaussmf | 2 | 1 addition<br>2 multiplication<br>1 exponential |
| Bell | Gbellmf | 3 | 2 additions<br>1 division<br>1 power |
| S-shaped | Smf | 2 | 4 comparisons<br>2 additions<br>2 multiplications |
| Triangular | Trimf | 3 | 4 comparisons<br>1 addition<br>1 multiplication |
| Trapezoidal | Trapmf | 4 | 5 comparisons<br>1 addition<br>1 multiplication |

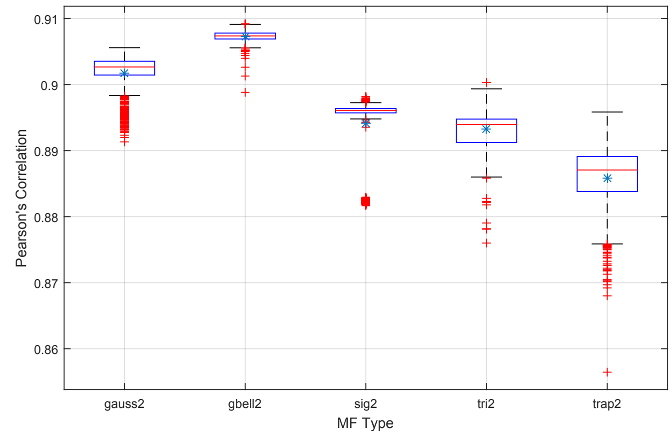



Fig. 8. Prediction performance for different types of MF.

demonstrates the box-plot of Pearson's correlation data generated during the validation phase for different types of MF. We can notice that the prediction performance of fuzzy IQA model related to bell MF is the best, since it has the best mean and the smallest standard deviation of the correlation scores. We can also notice that the fuzzy model related to S-shaped MF has a lot of abnormal points which are so close together, these produced abnormal results could be interpreted that the learning process, which is carried out to optimize the parameters of S-shaped MF model, is perhaps trapped into another local minimum and it cannot escape afterwards to go to the global minimum.

### *D. Number of rules (Number of MFs for each input)*

Fig. 9 illustrates the mean square errors obtained by the fuzzy networks in the training and validations phases; this figure shows that the training error mostly decrease as number of rules increases, while the validation error decreases up to a certain point (corresponding to number of rules equal to 12), and then it increases. This increase indicates an overfitting of the fuzzy model due to the use of too many rules. In fact, as the number of rules increases, the total number of adaptive parameters increases and the model will fit the training data points better, but it leads also to increase the both model complexity level and risk of overfitting.

With taking into account the assumptions mentioned previously in this paper, the total number of adaptive parameters (linear + nonlinear) is related to the number of rules as shows in the equations (13) and (14); where number of inputs is four and number of parameters in one MF is derived from Table I.

Table II shows the average and standard deviation (STD) of Pearson's correlation during the validation phase produce by different types and numbers of membership function. We can notice that for Gaussmf, Gbellmf and Smf functions the model performance mostly enhances as the number of MFs increase, in contrast, for Trimf and Trapmf functions the performance illogically declines, that is because the training phase of these two functions is not well developed inside ANFIS toolbox.

$$Number\ of\ linear\ parameters\ = (Number\ of\ inputs + 1) \times Number\ of\ rules \quad (13)$$

$$Number\ of\ nonlinear\ parameters\ = Number\ of\ inputs \times\ Number\ of\ rules\ \times Number\ of\ parameters\ in\ one\ MF \quad (14)$$

## V. Simulation Results

Having modified certain ANFIS settings and analyzed the prediction performance of the corresponding generated FIS models, we will select one fuzzy IQA model and compare its prediction performance and circuit complexity with other IQA models generated by different ML methods mentioned in Section III. The selected model consists of four input variables, each input has two generalized bell MFs, two fuzzy rules and one output (Fig. 10 and Fig. 4 present respectively fuzzy

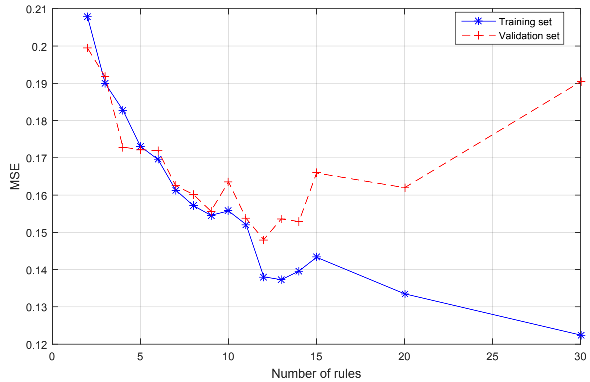



Fig. 9. Training and validation error curves for different number of rules.

TABLE II
AVERAGE – STD OF PLCC FOR DIFFERENT TYPE AND NUMBER OF MFS.

| Type \ Number | | 2 | 3 | 4 | 5 | 6 | 11 |
|---|---|---|---|---|---|---|---|
| Gaussmf | Mean | 90,17% | 90,89% | 91,42% | 91,67% | 91,90% | 92,70% |
| | STD | 0,29% | 0,11% | 0,19% | 0,16% | 0,15% | 0,16% |
| Gbellmf | Mean | **90,73%** | 91,13% | 91,42% | 91,63% | 91,85% | 92,70% |
| | STD | **0,08%** | 0,12% | 0,18% | 0,18% | 0,20% | 0,17% |
| Smf | Mean | 89,42% | 89,84% | 89,81% | 89,91% | 89,97% | 89,99% |
| | STD | 0,49% | 0,72% | 0,80% | 0,93% | 0,99% | 1,11% |
| Trimf | Mean | 89,32% | 88,37% | 87,16% | 86,83% | 86,15% | 80,15% |
| | STD | 0,27% | 0,98% | 1,03% | 1,40% | 0,93% | 2,28% |
| Trapmf | Mean | 88,58% | 86,03% | 83,78% | 83,01% | 81,98% | 71,71% |
| | STD | 0,50% | 2,10% | 1,44% | 1,13% | 1,68% | 2,70% |

inference diagram and architecture of the resulting model). This model has as total 34 adaptive parameters (10 linear and 24 nonlinear) which are optimized during the training phase with the following ANFIS settings: scatter partition strategy, initial step size equal to 0.5 and number of training epochs equal 23. Fig. 11 illustrates a high dependence between the predicted and subjective MOS scores during the validation phase. Looking back to Table I and the equation (9), we can see that each MF contains two additions, one division and one power units. The power block is considered the most complicated unit to be executed in integrated circuits, in particular if its power parameter (which is represented in the equation (9) by 2b) is real, not integer. Unfortunately all bell MFs in the produced model have real power parameter (2b) taken value in the range of 6.4 to 6.8. To simplify the circuit complexity of bell MF, the parameter b of all MFs is set to the same integer value. Table III shows the prediction performance of the fuzzy model in the case of: the parameter b is real (no change) and integer (equal to 4, 2 or 1). We can notice that replacing the real value of b with 4, the IQA model maintains mostly the same prediction performance, while replacing the real value of b with 2 or 1 the performance reduces and error increases. Furthermore, setting b to 4 (2 or 1) the power unit can be replaced by three (two or one respectively) successive multiplication units.

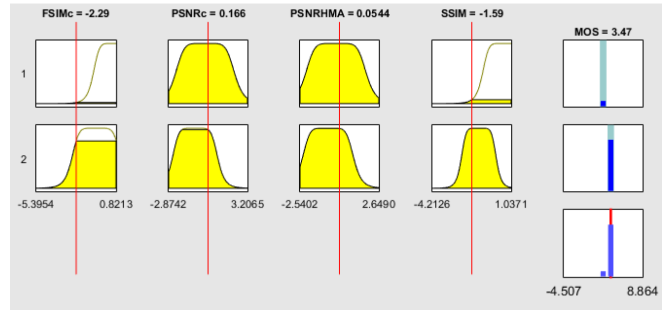



Fig. 10. Fuzzy inference diagram.

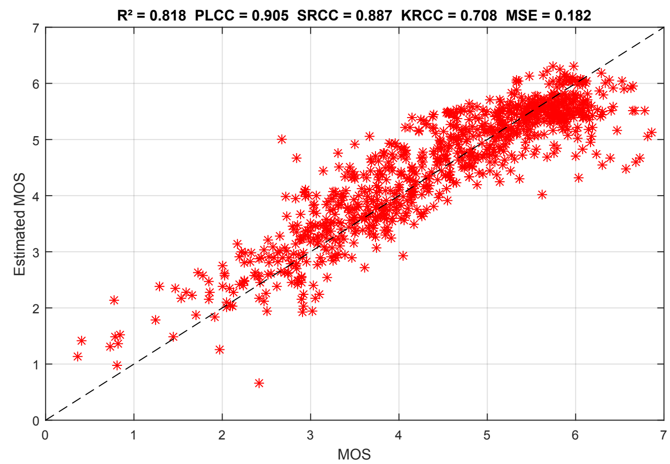



Fig. 11. MOS vs estimated MOS for the selected FL model.

Table IV and Fig. 12 compare the prediction performance of IQA model derived by using fuzzy logic approach to the models produced by the following five machine learning methods: LDA, k-NN, ANN, NLR and DT. The performance results are calculated during the validation phase by applying Monte Carlo Cross-Validation with 1000 cases. Table IV evaluates the means and standard deviations for several correlation scores (PLCC, SRCC, KRCC, $R^2$ and MSE), while Fig. 12 presents the boxplot of Pearson's correlation scores. The simulation results confirm that Fussy logic model is considered as the best in fitting the subjective MOS with high stable behavior, since it has the best mean and the smallest standard deviation of the correlation scores and the smallest MSE error.

TABLE III
PREDICTION PERFORMANCE FOR DIFFERENT B VALUE OF BELL MF.

| | **b is real** | **2b=8** | **2b=4** | **2b=2** |
|---|---|---|---|---|
| **MSE** | 0.182 | 0.182 | 0.202 | 0.239 |
| **$R^2$** | 0.818 | 0.819 | 0.798 | 0.761 |
| **PLCC** | 0.905 | 0.905 | 0.895 | 0.881 |
| **SRCC** | 0.887 | 0.888 | 0.882 | 0.871 |
| **KRCC** | 0.708 | 0.708 | 0.700 | 0.685 |
| # multi. | — | 3 | 2 | 1 |

TABLE IV
PERFORMANCE (AVERAGE – STD) OF DIFFERENT ML METHODS.

| | | PLCC | SRCC | KRCC | $R^2$ | Error |
|---|---|---|---|---|---|---|
| LDA | Mean | 82,13% | 81,39% | 73,93% | 66,05% | 33,95% |
| | STD | 0,88% | 0,51% | 0,54% | 1,83% | 1,83% |
| KNN | Mean | 86,12% | 85,99% | 79,14% | 72,74% | 24,26% |
| | STD | 0,82% | 0,90% | 1,03% | 1,65% | 1,65% |
| ANN | Mean | 88,65% | 87,31% | 69,41% | 78,47% | 21,53% |
| | STD | 0,55% | 0,49% | 0,56% | 1,54% | 1,54% |
| NLR | Mean | 89,99% | 88,30% | 70,63% | 80,91% | 19,09% |
| | STD | 0,24% | 0,25% | 0,33% | 0,44% | 0,44% |
| DT | Mean | 90,02% | 89,20% | 73,66% | 80,91% | 19,09% |
| | STD | 0,58% | 0,57% | 0,76% | 1,07% | 1,07% |
| FL | Mean | **90,73%** | **89,04%** | **71,25%** | **82,24%** | **17,76%** |
| | STD | **0,08%** | **0,12%** | **0,20%** | **0,15%** | **0,15%** |

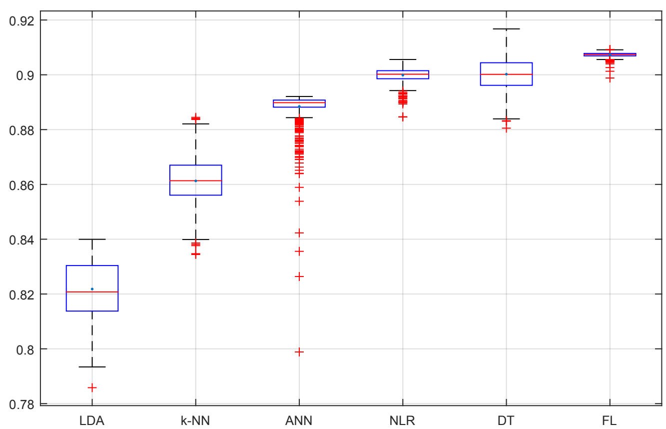



Fig. 12. Performance comparison for different ML methods.

## VI. CONCLUSION

In this paper, we studied the methodology of IQA process based on fuzzy logic. We used the image quality database TID2013 to benchmark and evaluate the generated models. In order to improve model training performance and to avoid redundancy problem, we selected the most significant image quality metrics based on Pearson's correlation measure. Several structure of fuzzy IQA models were produced by changing certain ANFIS configurations, then we selected the best model, which has the good compromise between a good prediction accuracy and low implementation complexity. The final architecture of the selected FL model consists of two bell-shaped MFs associated to each input metric, two fuzzy if-then rules and two linear combination equations. Finally we compared the performance of the produced FL model with the models produced by the other IQA methods (LDA, k-NN, ANN, NLR and DT), and we found out that FL model provides the best prediction accuracy and stability. Future work will be to implement the FL IQA model inside the integrated circuits to validate its behavior on the real runtime conditions.

## REFERENCES

[1] N. Ponomarenko, O. Ieremeiev, V. Lukin, L. Jin, K. Egiazarian, J. Astola, B. Vozel, K. Chehdi, M. Carli, F. Battisti *et al.*, "A new color image database tid2013: innovations and results," in *Advanced Concepts for Intelligent Vision Systems*. Springer, 2013, pp. 402–413.
[2] H. R. Wu, W. Lin, and L. J. Karam, "An overview of perceptual processing for digital pictures," in *Multimedia and Expo Workshops (ICMEW), 2012 IEEE International Conference on*. IEEE, 2012, pp. 113–120.
[3] A. K. Moorthy and A. C. Bovik, "Visual quality assessment algorithms: what does the future hold?" *Multimedia Tools and Applications*, vol. 51, no. 2, pp. 675–696, 2011.
[4] L. Xu, W. Lin, and C.-C. J. Kuo, *Visual quality assessment by machine learning*. Springer, 2015.
[5] R. Alhakim, G. T. Tchendjou, E. Simeu, and F. Lebowsky, "Image quality assessment using nonlinear learning methods," in *2015 27th International Conference on Microelectronics (ICM)*. IEEE, 2015, pp. 5–8.
[6] R. Fang, R. Al-Bayaty, and D. Wu, "Bnb method for no-reference image quality assessment."
[7] B. Akoa, E. Simeu, and F. Lebowsky, "Using artificial neural network for automatic assessment of video sequences," in *Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on*. IEEE, 2013, pp. 285–290.
[8] B. E. Akoa, E. Simeu, and F. Lebowsky, "Video decoder monitoring using non-linear regression," in *On-Line Testing Symposium (IOLTS), 2013 IEEE 19th International*. IEEE, 2013, pp. 175–178.
[9] A. C. Cameron and F. A. Windmeijer, "An r-squared measure of goodness of fit for some common nonlinear regression models," *Journal of Econometrics*, vol. 77, no. 2, pp. 329–342, 1997.
[10] "Tid2013 database: http://www.ponomarenko.info/ tid2013.htm."
[11] N. Ponomarenko, L. Jin, O. Ieremeiev, V. Lukin, K. Egiazarian, J. Astola, B. Vozel, K. Chehdi, M. Carli, F. Battisti *et al.*, "Image database tid2013: Peculiarities, results and perspectives," *Signal Processing: Image Communication*, vol. 30, pp. 57–77, 2015.
[12] G. T. Tchendjou, R. Alhakim, E. Simeu, and F. Lebowsky, "Evaluation of machine learning algorithms for image quality assessment." IEEE, 2016.
[13] Q.-S. Xu and Y.-Z. Liang, "Monte carlo cross validation," *Chemometrics and Intelligent Laboratory Systems*, vol. 56, no. 1, pp. 1–11, 2001.
[14] G. McLachlan, *Discriminant analysis and statistical pattern recognition*. John Wiley & Sons, 2004, vol. 544.
[15] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.
[16] L. Rokach and O. Maimon, *Data mining with decision trees: theory and applications*. World scientific, 2014.
[17] J.-S. R. Jang and C.-T. Sun, "Neuro-fuzzy modeling and control," *Proceedings of the IEEE*, vol. 83, no. 3, pp. 378–406, 1995.
[18] B. Khoshnevisan, S. Rafiee, M. Omid, and H. Mousazadeh, "Development of an intelligent system based on anfis for predicting wheat grain yield on the basis of energy inputs," *Information processing in agriculture*, vol. 1, no. 1, pp. 14–22, 2014.
[19] Y. H. Joo and G. Chen, "Fuzzy systems modeling: an introduction," *Encyclopedia of artificial intelligence*, 2009.
[20] L. K. PinPin, D. F. T. Gamarra, C. Laschi, and P. Dario, "Topology based fuzzy clustering for robust anfis creation," in *Cybernetic Intelligent Systems, 2008. CIS 2008. 7th IEEE International Conference on*. IEEE, 2008, pp. 1–6.
[21] B. Fritzke, "Incremental neuro-fuzzy systems," in *Optical Science, Engineering and Instrumentation'97*. International Society for Optics and Photonics, 1997, pp. 86–97.
[22] A. Al-Hmouz, J. Shen, R. Al-Hmouz, and J. Yan, "Modeling and simulation of an adaptive neuro-fuzzy inference system (anfis) for mobile learning," *Learning Technologies, IEEE Transactions on*, vol. 5, no. 3, pp. 226–237, 2012.
[23] J.-S. R. Jang, "Anfis: adaptive-network-based fuzzy inference system," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 23, no. 3, pp. 665–685, 1993.
[24] S. Sivanandam, S. Sumathi, S. Deepa *et al.*, *Introduction to fuzzy logic using MATLAB*. Springer, 2007, vol. 1.