The paper proposed by Adrià Amell and colleagues presents an inversion technique based on machine learning for the estimation of ice wather path (IWP) form Meteosat-9 observations with a focus on low latitudes. In their work, the authors both introduce and describe the topic with good details and discuss the potential and advantages of using artificial intelligence quantile-based regression methodologies over physics-based methods present in the literature.

In this context, the authors test various neural network architectures and compare the use of observations in the thermal infrared (IR) and/or visible bands as inputs. Finally, authors conclude that the architecture based on convolutional neural networks (CNNs) in which spatial information is integrated is the architecture that performs better, using, moreover, only observations in the infrared band as input. The presented approach offers several advantages over traditional methods, such as the ability to calculate diurnal cycles, a problem that for example CloudSat cannot solve due to its limited temporal and spatial sampling. Then, since the methodology is quantile based, it allows the developed methodology to obtain directly and in an integrated way an estimate of the uncertainty of the regressions.

The authors validated their work using CLASS that is thoroughly validated dataset based on traditional approaches. The obtained retrievals compare favourably with IWP retrievals in CLAAS. In my opinion, this last result arguably demonstrates the potential of this methodology highlight the possibilities to overcome limitations from physics-based approaches as demonstrated in other works recently published in literature Holl et al. (2014), Islam and Srivastava (2015) and Mastro et al. (2022).

However, in my opinion, some shortcomings are present in the paper framework that require a major review.

1) In section 3.2 authors describe the Network architecture and specifically they discuss the multilayer percepton (MLP) and the CNN configurations indicating their structural hyperparameters. I would argue that it is essential to describe in more detail this information and how the choice of these configurations was made. For example, for the MLP configuration, the authors indicate an architecture consisting of 16 hidden layers each composed of 128 hidden units assuming that it is the setup that achieves the best performance. How did they reach this finding? Has a tuning framework been used? If so, how was the hyperparameter space configured from which to begin the search for the best configuration? Also, were configurations with fewer hidden layers explored?

2) The authors indicate that Table 2 shows the input characteristics used by the analyzed architectures. I believe that as presented, the table does not make it easy to understand which of the inputs shown are used of the architectures presented. I understand that various configurations of inputs were used for each architecture. Anyway, I sugges the authors reformulate more clearly the information in Table 2 and contextualize it better.

3) In section 3.3 the authors discuss the training of the proposed configurations. Here they also introduce information regarding the inputs used. In general as presented the section is very confusing and a possible reader might find it difficult to read. I propose to move the choice of inputs to section 3.2 following the corrections of Table 2 indicated previously and to focus section 3.3 in providing details concerning only the training phase. In addition, a useful piece of information would be to show the learning curves (for each epoch of training and

validation) of the two configurations in order to demonstrate the absence of overfitting and underfitting problems.

4) Figure 4 shows the CNN architecture and in my opinion it is a bit misleading. I would like to propose to the authors to change the position of the DXception and Xception blocks next to the blocks themselves, because as they look they appear to be part of the input and output blocks.