The main issue with this paper is that the proposed method and metrics miss the context in which a file type detection tool would be deployed and thus the results seem not to be answering the right question. Focusing on true positive/false positive rate of file detection implicitly assumes that having encrypted files (of a certain kind) is indicative of being a criminal. There is nothing in the paper that shows this is the case:

* First, the authors claim that there is an "alarming" (it would be better to avoid judgemental adjectives and stick to facts) growth of the use on encryption in crime.
  + Removed ‘alarming’
* While numbers support the observation that there is an increase in criminal use of encryption, this is only relevant if this increase is larger than the use of encryption by non-criminal users. The authors need to establish a baseline of non-criminal use of encrypted files. If the use of encryption is similar in both criminal and non-criminal populations, the output of the tool is then not necessarily an indicator of crime. False positives with respect to crime content will occur not only because the ML detector is inaccurate, but also because the detector flags non-criminal content. As the majority of people are non-criminals, this would likely result on a large number of cases to be checked manually with no gain.
  + The aim of eDefenders is to selectively remove a subset of people being investigated that are in possession of file-types that pose no ‘threat’. If 10 people are being investigated for CSAM, and 4 of them are in possession of solely doc, docx, and txt files then eDefenders will have reduced the population size by 40% that law enforcement agents are required to allocate resources toward for further investigation. Additionally, eDefenders supports a human-in-the-loop system to be used holistically with other factors (shown in Fig 1) such as criminal background. This means it will not be deployed on every person that comes through the border or every person with a computer, it is suggested to be used with people who have already exhibited patterns of criminality and LEA needs to decide if there is reason to pursue further investigation.

The population it will be used on will already have been filtered to changed the characteristics compared to the general population.

* Second, the metrics the authors use implicitly assume that having a certain quantity of files of a particular type is indication of being a criminal. This is also not supported by facts. To make a claim that having directories with large number of encrypted images/videos is a sign of criminal activity, the authors need again a baseline of how the distribution of file types is in the non-criminal population. To give some examples, users of the Signal messenger are likely to have large folders of images and videos coming from friends and family; high executives are likely to have large folders of encrypted business-sensitive documents; and investigative journalists are likely to have large volumes of encrypted information from all types. Without a clear baseline of the difference in file types distribution in criminal and non-criminal populations, it is not possible to define a false positive rate.
  + <https://commons.erau.edu/cgi/viewcontent.cgi?article=1472&context=jdfsl>
  + There is precedence for LEA needing file types to support digital forensic task. This can be seen in discussion our group had with active LEAs and in the digital forensic literature.
* Third, there is no support that the experimental setup of the authors is representative of what a tool like eDefender would encounter in the wild. The issue is two fold: the chosen transformation functions and the The authors decide to use AES-CBC-256 and WinRar as algorithms to create the dataset. But they do not discuss if these are representative. In the case of encryption, since it seems that we know which tools criminals use, why not use those tools? At the least I would expect a study that shows that one-time encryption of files or one-time encryption and compression results in a BFD similar to that output by tools like BitLocker or TrueCrypt -- which to my understanding do more complex operations to manage encrypted disks.
  + I agree this is interesting work that would contribute to the field. However, the work that is presented is iterative, and the encryption algorithms used were to determine if our proposed method was even viable for further consideration. Future work will include more diverse encryption algorithms.
* Fourth, seeing all points above in conjunction, another question is not only if the use of encryption is generalized, but which tools are generally used. It may be the case that while encryption in general has grown similarly in the criminal and non-criminal domains, the tools are different. At that point, what would maybe be the way to flag potential cases for LEAs would not be detecting file types, but the existence of encryption using a certain software -- not studied in this paper.
* (left blank intentionally – open for discussion)

A no less important issue than the misalignment of method and objective, is that the paper's ethics discussion does not speak about the potential for harm that a tool such as eDefender has. The authors must include an analysis of the dangers that introducing such a tool could have for potential surveillance or monitoring of non-criminal activities (e.g., journalists devices being seized at the border when leaving countries governed by oppressive regimes). I note that human intervention is not of any help when the tool itself is being misused to detect activities other than crime as those able to do the checks are the ones that have decided on misuse.

* The purpose of eDefender is to provide additional insights without impeding privacy. There is no point in the process where encryption is broken. Therefore, a journalist’s device being investigated with eDefenders would provide no value to an oppressive regime. This is also why we believe it supports American constitutional rights.

The paper would also greatly benefit from doing deeper studies on the patterns detected and the distribution of errors. For example, in Section 5.2, authors talk about "factor" and "clusters", but do not explain what these are. What is giving away the different encryption/compressed file types? What patterns are actually preserved? are they preserved equally across all transformations? The latter question is really important, as all numbers are reported globally. It may be the case that the tool does exceptionally well in plain and compressed text (not surprising), but much worse in encryption -- the actual goal.  
Also results are mostly presented stratified by file type, which is interesting, but does not necessarily reflect/explain the fact that these types can be separated. Are the factors/features that can be detected in pdf the same as in text? what is the difference? A deeper analysis of why and when the method would work would provide a much stronger support for the conclusions of the paper that the current obscure analysis.

Minor comments and questions:

* Was there any manual check to validate that the search terms in Section 4.2 actually reflect crimes in which encryption was a hindrance for the investigation? or can it be the case that encryption is mentioned for different reasons?
* Algorithm 1 and 2 do not render well and thus are hard to follow
* Figure 4 might be better presented as a Precision-Recall curve that allows to better understand the trade-offs of the thresholds. Having to compare cells in the grids is not very practical.
* Figure 5 clearly shows a decreasing trend, but this trend also flattens out in 60, which I interpret as False positives cannot be fully removed. What are the implications of this? Also it seems that the Y axis is given in absolute numbers. What is the percentage on the dataset? It is needed to grasp the actual volume when the tool is deployed at scale.
  + Adjusted caption to say “Relationship of Uncertainty Threshold vs number of False Positives (FP) samples detected in Testing Data (20\% of total data)”