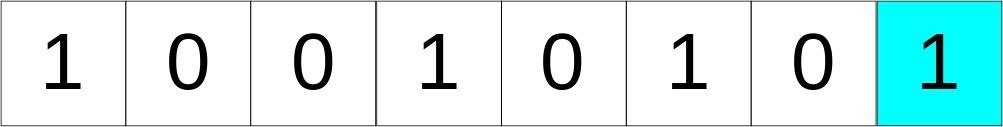
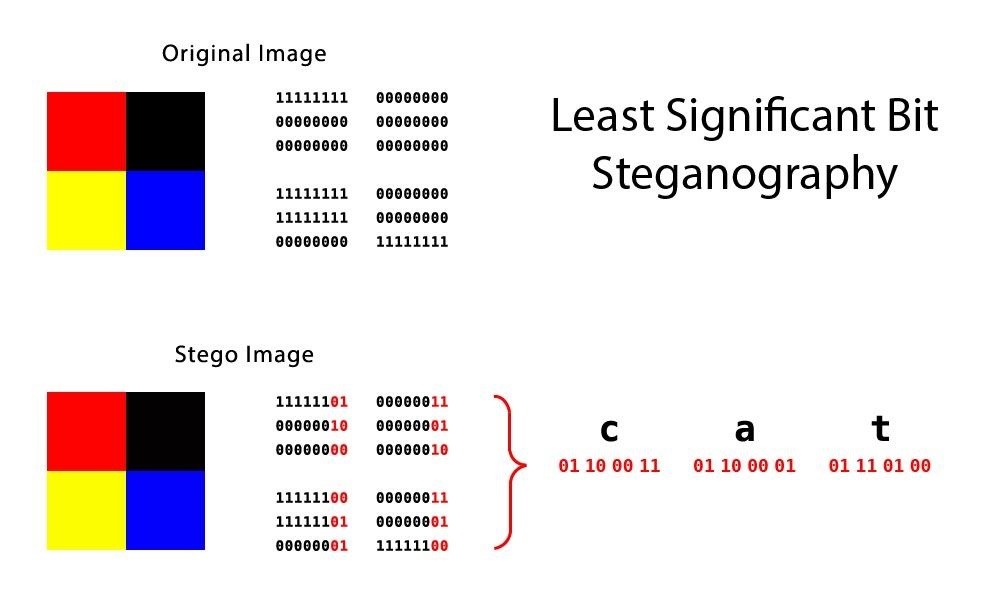
Einleitung

Steganographie wird schon seit vielen Jahrhunderten eingesetzt um wichtige Nachrichten zu verstecken, sodass diese nicht bemerkt werden. Der frühste dokumentierte Einsatz fand bereits im Jahre 440 v.C. statt, als ein persischer Herrscher seinem treusten Diener eine Nachricht auf die Kopfhaut tätowieren lies und ihn zu seinem Vetter schickte, als seine Haare nachgewachsen waren. Im Jahre 1966 hat ein amerikanischer Kriegsgefangener der Nord-Vietnamesen bei einer Pressekonferenz mit wiederholtem Blinzeln in Morse-Code die Nachricht „F-O-L-T-E-R“ buchstabiert und damit bestätigt, dass amerikanische Kriegsgefangen gefoltert werden. Im Laufe der Jahre wurden viele kreative Wege dokumentiert eine Nachricht hinter etwas anderem zu verstecken. In den folgenden Kapiteln werden die wichtigsten digitale Methoden der Steganografie vorgestellt.

LSB

Eine grundlegende Mechanik der digitalen Steganografie ist das Verstecken einer Nachricht in den Least-Signifikant-Bits einer Datei. Dazu wird zum Beispiel in einem Byte immer das LSB entsprechend einer Nachricht in Binärcode verändert, wie in Abbildung 1 zu sehen.

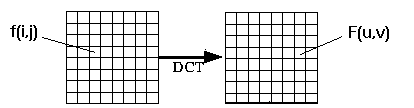
Wenn die Datei ein Bild oder eine Audiodatei ist, dann wird die Veränderung den menschlichen Sinnen nicht auffallen, sondern lediglich ein geringes Rauschen verursachen. Bei einer Textdatei würde es auffallen, da sich hier Zeichen entsprechend der Codierung verändern würden. Es können, wie im folgenden Beispiel, auch mehrere niederwertige Bits verändert werden, wodurch jedoch das Rauschen größer werden würde.



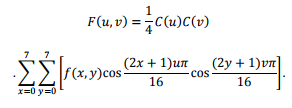
In Abbildung 2 ist ein beispielhaftes Bild mit 4 Pixeln dargestellt. Jedes Pixel hat eine Farbtiefe von 8-bit und benötigt damit 3 Byte für die Codierung der RGB-Farbe. Wenn die 2-niederwertigsten Bits für das Verstecken einer Nachricht benutzt werden, dann ergibt sich für die Nachricht eine Speichergröße von 3 Byte.

DCT

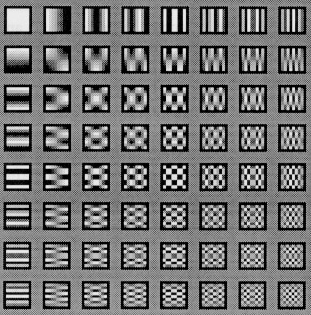
Die Diskrete Cosinus Transformation ist eine spezielle Fouriertransformation – Das Eingangsbild wird also vom Bildbereich in den Zeitbereich transformiert. Der Algorithmus hilft dabei ein Bild in Teile unterschiedlicher Wichtigkeit aufzuteilen und damit zu errechnen, wo Veränderungen das geringste Rauschen verursachen.



Das Eingangsbild wird dafür in 8x8-Matrizen aufgeteilt und daraufhin wird für jeden Index i und j einer Teilmatrize die Transformation aus Abbildung 4 durchgeführt:



Um die DCT-Koeffizienten zu errechnen werden alle Frequenzanteile, zu sehen in Abbildung 5 summiert. Dabei wächst die Frequenz vom linken oberen Eck der DCT nach rechts unten. Bei den meisten Bildern liegt ein großer Anteil der Information in den niedrigen Frequenzen, während die hohen Frequenzen wenig Informationen haben. Deshalb wird die geheime Nachricht hier versteckt. Zuletzt wird nach dem Hinzufügen der Daten eine Rücktransformation durchgeführt.



Append Data

Eine weitere sehr einfache, aber gebräuchliche Methode ist es Daten an eine Datei anzuhängen, sodass sie von einem Leseprogramm abgeschnitten werden. Bei Bildern können zum Beispiel einfach Daten an das Ende gehängt werden, sodass sie von Bildprogrammen nicht angezeigt werden. So sind die Daten visuell nicht erkennbar, die Dateigröße verändert sich jedoch, was sehr einfach bemerkt werden kann. Andere digitale steganographische Methoden haben den Vorteil, dass die Dateigröße nicht verändert wird, wodurch sie erst durch genauere Analyse enttarnbar sind. Dafür ist das Anhängen von Daten jedoch sehr einfach implementierbar und benötigt nur oberflächliche Programmierkenntnisse.

<https://en.wikipedia.org/wiki/Steganography>

<http://www.mathe.tu-freiberg.de/~hebisch/cafe/kryptographie/steganographie.html>

<https://null-byte.wonderhowto.com/how-to/introduction-steganography-its-uses-0155310/>

<https://users.cs.cf.ac.uk/Dave.Marshall/Multimedia/node231.html>