Linear Program

November 2024

1 Specification

Let P be the set of all pixels and $n = |P|$. Let S be a set of all possible clusters of pixels and $m = |S|$. For every $i = 1, 2, ..., m$ we will denote $C_i \in S$ as a binary vector of length n (i.e. $C_i \in \{0,1\}^n$), with:

$$[C_i]_j = \begin{cases} 1 & \text{if pixel } j \text{ is in cluster } C_i \\ 0 & \text{otherwise} \end{cases}$$

Let $C \in \{0,1\}^{m \times n}$ be a matrix with $C_{ij} = [C_i]_j \quad \forall 1 \le i \le m, 1 \le j \le n$.

Thus, we can infer that $m = 2^n$. Let's say we precomputed the cost of each cluster and stored it in a vector $D \in \mathbb{R}^m$ with $D_i = cost(C_i)$.

Let $\beta \in \{0,1\}^m$ and β_i is set as 1 if and only if C_i is one of the clusters of the clusterization K.

Now, we can define our Integer Problem for k-clusterization.

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} D_i \beta_i \\ \text{s.t.} \quad & \sum_{i=1}^{m} \beta_i = k && (1) \\ & \sum_{i=1}^{m} \beta_i C_{ij} = 1 \qquad \forall j = 1, \dots, n && (2) \\ & \beta_i \in \{0,1\} \qquad \forall i = 1, \dots, m \end{aligned}$$

Condition (1) ensures that there are exactly k clusters. Condition (2) ensures that each point belongs to exactly one cluster.

P.S.

Lemma: Any two pixels of the same color end up in the same cluster under the optimal clusterization.

We run local search initialized at 100 different random seed and about an upper bound M for optimal clusterization cost.

Then, I use the lemma to find all possible clusters C_i for $i \in [q]$ with a cost less than M. I can do that since I can treat work with unique colors instead of all pixels and apply backtracing with an upper bound on the cost of a cluster. Each cluster will be defined as a set of colors in the lemma. Thus, n in this application would be equal to the number of unique colors, i.e. 20.

Then, I store cluster costs in $D \in \mathbb{R}^q$ and solve the IP from problem 2.