

Bayessche Texturanalyse  
Seminar: Bayessche Ansätze in der Bildanalyse

Patrick Eisele

24.Juli 2006

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>3</b>
1.1	Textur . . . . .	3
1.2	Textur-Analyse . . . . .	3
<b>2</b>	<b>Textur-Modellierung</b>	<b>4</b>
2.1	Grundlagen . . . . .	4
2.2	$\psi$ -Modell . . . . .	5
<b>3</b>	<b>Textur-Segmentierung</b>	<b>6</b>
3.1	Grundlagen . . . . .	6
3.2	Bayessche Textur-Segmentierung . . . . .	6
<b>4</b>	<b>Textur-Klassifizierung</b>	<b>9</b>
4.1	Grundlagen . . . . .	9
4.2	Kontextuelle Bayessche Textur-Klassifizierung . . . . .	10
4.3	Nicht-kontextuelle Bayessche Textur-Klassifizierung . . . . .	11
<b>5</b>	<b>Anhang</b>	<b>12</b>
5.1	Literaturangaben . . . . .	12

# 1 Einleitung

## 1.1 Textur

Der Begriff Textur wird in vielen verschiedenen wissenschaftlichen Disziplinen verwendet. Jedoch gibt es keine einheitliche Definition dieses Begriffes. Definitionen sind je nach Anwendungsgebiet und gewähltem Ansatz unterschiedlich. Daher sollen hier zwei gängige Definitionen aus dem Bereich der Bildanalyse gegeben werden:

**Definition 1:**

Textur ist ein Attribut, das die räumliche Anordnung der Grauwerte in einer Region darstellt.

**Definition 2:**

Textur ist eine visuelle Charakteristik eines Bildsegmentes, die das Segment einer bestimmten Klasse zuordnet. Diese Klassen sind oft mit physischen Interpretationen verbunden (z.B. Holz, Gras, etc. ).

## 1.2 Textur-Analyse

Ziel der Texturanalyse ist es, mit Hilfe geeigneter Parameter Texturen zu beschreiben und erzeugen, Texturen zu unterscheiden und sie zu klassifizieren.

Die Texturanalyse lässt sich daher in folgende drei Disziplinen einteilen: **Textur-Modellierung**, **Textur-Segmentierung**, **Textur-Klassifizierung**.

Die **Textur-Modellierung** dient dabei der stochastischen Beschreibung und Erzeugung von Texturen mit Hilfe geeigneter Parameter.

Die **Textur-Segmentierung** dient dazu, Bildsegmente mit unterschiedlicher texturaler Ausprägung voneinander abzugrenzen.

Die **Textur-Klassifizierung** dient zur Erkennung von bestimmten Texturen und deren Einordnung in vordefinierte Klassen.

## 2 Textur-Modellierung

### 2.1 Grundlagen

Bei der Textur-Modellierung geht es darum Modelle zu erstellen. Mit Hilfe dieser Texturmodelle sollen Texturen beschrieben und erzeugt werden. Im weiteren Verlauf wird hier ein stochastischer Ansatz zur Textur-Modellierung vorgestellt. Im stochastischen Zusammenhang versteht man unter einem Texturmodell eine parametrische Familie von Zufallsfelder, die von einer gewissen Anzahl an Hyperparametern abhängt. Eine spezifische Textur in einer solchen Familie, bzw. bei einem solchen Texturmodell, ist durch die Wahl der Hyperparametern charakterisiert.

Bei der Wahl eines Texturmodelles zur Beschreibung/ Erzeugung von Texturen sind zwei Aspekte zu beachten:

1. Angemessene Modellklasse wählen
2. Passende Hyperparameter bestimmen

Ziel der Textur-Modellierung wird also sein, Texturen stochastisch mit Hilfe eines Gibbs-Zufallsfeldes zu beschreiben.

**Definition:**

Ein Zufallsfeld  $Y$  ist ein Gibbs-Feld, falls dessen Verteilung folgende Darstellung hat:

$$\Pi(y) = \frac{1}{Z} \exp\{-K(y)\} \quad (1)$$

Daher gilt es, eine geeignete Energiefunktion  $K(y)$  zu bestimmen, um Texturen zu beschreiben.

Im weiteren Verlauf soll eine Modellklasse näher beschrieben werden : **Das  $\Psi$ -Modell**

**Nachbarschaftsbeziehungen:**

Um Texturmodelle genauer vorstellen zu können, sollen zunächst einmal Nachbarschaftsbeziehungen, bzw. Annahmen über deren Eigenschaften betrachtet werden. Wie bei den meisten Texturmodellen wird auch hier Translations-Invarianz vorausgesetzt. Dies bedeutet, dass jede Nachbarschaft  $\partial\{s\}$  von  $s \in Z^2$  eine Verschiebung  $s + \partial\{0\}$  einer Nachbarschaft  $\partial\{0\}$  von 0 darstellt und es gilt:

$$\Pi(Y_s = y_s \mid Y_{\partial(s)} = y_{\partial(s)}) = \Pi(Y_t = (\phi_{t-s}(y))_t \mid Y_{\partial(t)} = (\phi_{t-s}(y))_{\partial(t)}) \quad (2)$$

wobei  $(\phi_u(y))_t = y_{t-u}$  die Verschiebung darstellt.

Seien daher  $C_i$  ( $i = 1, \dots, d$ ) Paarcliquen, die  $0 \in Z^2$  enthalten, so definieren diese  $C_i$  Nachbarschaftsbeziehungen  $t \stackrel{i}{\sim} s$  vom Typ  $i$  in dem Sinne, dass  $s, t$   $i$ -Nachbarn sind, falls  $\{s, t\}$  eine Translation von  $C_i$  ist.

## 2.2 $\psi$ -Modell

Das  $\psi$ -Modell, das im Folgenden vorgestellt werden soll, ist nur eines von vielfältigen Modellen zur Textur-Modellierung. Diese Modelle unterscheiden sich jeweils in der Wahl der Energiefunktion, die das zugehörige Gibbs-Feld beschreibt. Die Energiefunktion des  $\psi$ -Modell hat folgende Form:

$$K(y) = \sum_{i=1}^d \vartheta_i \sum_{t \sim_s^i} \psi(y_s - y_t) \quad (3)$$

wobei  $\psi(\omega)$  eine Abweichungsfunktion, die symmetrisch um 0 und steigend in  $|\omega|$  ist. Durch diese Funktion lässt sich die Abweichung der Intensitäten benachbarter Pixel verschiedentlich bestrafen. Dadurch lassen sich erwünschte Eigenschaften bezüglich der Übergänge bevorzugen.

Wahl von  $\psi$  als flache Tassenfunktion  $\Rightarrow$  Texturen mit fließenden Übergängen

Wahl von  $\psi$  als steile Parabel  $\Rightarrow$  Texturen mit klaren Übergängen

$\vartheta_i \in \mathcal{R}$  sind die Gewichtungen der  $i$ -Nachbarschaften. Durch diese Gewichtungen lässt sich die Korrelation zwischen den Intensitäten von Pixeln, die  $i$ -Nachbarschaften unterhalten, bestimmen.

Wahl von positiven  $\vartheta_i \Rightarrow$  Gleiche Intensitäten der Nachbarn bevorzugt

Wahl von negativen  $\vartheta_i \Rightarrow$  Unterschiedliche Intensitäten bevorzugt

Wahl von kleinen Werte  $|\vartheta_i| \Rightarrow$  schwaches „Coupling“

Wahl von großen Werte  $|\vartheta_i| \Rightarrow$  starkes „Coupling“

### Anmerkungen:

Eine spezielle Textur innerhalb eines Textur-Modelles ist durch Wahl der Nachbarschaftsbeziehung und deren Gewichtungen  $\vartheta_i$  vollständig charakterisiert.

Um angemessene Hyperparameter  $\vartheta_i$  zu bestimmen, wird häufig das Prinzip des „*supervised learning*“ genutzt(d.h. die Parameter werden mit Hilfe von Ausschnitten, die reine Textur enthalten, geschätzt).

Es existieren zahlreiche Textur-Modelle, die jeweils spezielle Eigenschaften von Texturen widerspiegeln.

All diese Modelle hängen von einer gewissen Anzahl an Hyperparametern ab.

## 3 Textur-Segmentierung

### 3.1 Grundlagen

Texturen sind durch eine gewisse Anzahl an Eigenschaften charakterisiert. Der Ansatz bei der Textur-Segmentierung ist nun, Texturen mit Hilfe einer minimalen Anzahl an charakteristischen Eigenschaften zu unterscheiden. Meist sind das einfache Eigenschaften der Intensitätsstruktur. Texturen werden dann als unterschiedlich bewertet, falls sie in mindestens einer dieser charakteristischen Eigenschaft merklich voneinander abweichen.

Dieser Ansatz kann nun folgendermaßen formalisiert werden:

Sei  $(y_s)_{s \in S}$  eine Intensitätskonfiguration auf einem endlichen Gitter  $S$ . Seien  $B$  und  $D$  Regionen von Pixeln und  $L$  eine Menge von Labels  $l$ .

Diese Labels  $l$  dienen dazu, die Regionen auszuzeichnen, je nachdem welche Textur sie aufweisen, d.h.:

Falls Regionen gleichartige (bzw. unterschiedliche) gleichartige Textur  $\Rightarrow$  Regionen bekommen gleiches (bzw. unterschiedliches) Label.

Nun gilt es geeignete Abbildungen zu finden, welche die Intensitätskonfigurationen in den Regionen nach diesen charakteristischen Eigenschaften abbilden:

$$y_D \mapsto \phi^{(i)}(y_D) \in \Phi^{(i)} (= R^d) \quad (4)$$

Mit Hilfe dieser Abbildungen lässt sich nun eine starre Bedingung für Gleichheit formulieren:

$$d^{(i)}(\phi^{(i)}(y_B), \phi^{(i)}(y_D)) \leq c^{(i)} \quad \forall i \quad (5)$$

wobei  $c^{(i)}$  Schranken und  $d^{(i)}$  ein Abstand auf  $\Phi^{(i)}$  darstellen. Falls diese Bedingung nun erfüllt ist, bekommen die Regionen  $B$  und  $D$  gleiche Labels  $l_B = l_D$ .

### 3.2 Bayessche Textur-Segmentierung

Nun soll über diese starre Betrachtung hinausgegangen werden, und es soll ein stochastischer, bzw. bayesscher Ansatz zur Textur-Segmentierung vorgestellt werden. Im folgenden wird die Beschreibung eines Bayesschen Ansatzes zur Textursegmentation nach *D. Geman* näher betrachtet.

Ziel ist es eine Energiefunktion in folgender bekannter Form aufzustellen:

$$H(y, l) = K_{PL}(y, l) + V_L(l) \quad (6)$$

wobei  $K_{PL}(y, l)$  die Energiefunktion der Pixel-Label-Interaktion ist, mit Hilfe derer erwünschte Intensitätskonfigurationen der Pixel bezüglich einer speziellen Textur bevorzugt, bzw. unerwünschte bestraft werden können.

Und  $V_L(l)$  ist die a priori-Energie der Label-Label-Interaktion ist, mit Hilfe derer erwünschte Labelkonfigurationen der Regionen in  $S$  bevorzugt, bzw. unerwünschte bestraft werden können.

Durch Minimieren dieser Energiefunktion erhält man ein optimales Labelling der Regionen, bzw. eine optimale Segmentierung.

Die Voraussetzungen für diesen Ansatz seien folgende: Es sei  $S$  ein Quadratisches Gitter, mit  $S = \{(j, k) : 1 \leq j, k \leq n\} \subset Z^2$  mit einer endlichen Menge an Intensitäten  $G$

Desweiteren sei  $U$  ein Teil-Gitter von  $S$ , d.h. die Menge der Pixel von  $U$  ist eine Teilmenge der Pixel von  $S$  und es gilt:

$$U = U_\varrho = \{(j\varrho + 1, k\varrho + 1) : 0 \leq j, k \leq (n - 2)/\varrho\}, \quad \varrho \in Z \quad (7)$$

Dadurch wird das Gitter  $S$  gleichmäßig in Pixel-Blöcke  $D_u \subset S$  aufgeteilt, deren Zenrum das Pixel  $u \in U$  ist. Desweiteren seien  $(y_s)_{s \in S}$  ein Pixelprozess und  $(l_u)_{u \in U}$  ein Labelling

### Pixel-Label-Interaktion:

Die Pixel-Label-Interaktion kann folgendermaßen beschrieben werden:

$$K_{PL}(y, l) = \sum_{u \stackrel{\varrho}{\sim} v} \Psi_{uv}(y) \Phi_{uv}(l) \quad (8)$$

wobei  $u \stackrel{\varrho}{\sim} v$  bedeutet, dass  $u$  und  $v$  Nachbarn in  $U_\varrho$  sind.

$\Phi_{uv}(l) = \delta_{l_u, l_v}$ , d.h. falls die benachbarten Regionen mit dem gleichen Label ausgezeichnet werden, ist der Wert 1 und somit wird die Funktion  $\Psi_{uv}$  gewertet. Falls die benachbarten Regionen mit unterschiedlichen Labels ausgezeichnet werden, ist der Wert 0 und somit wird die Funktion  $\Psi_{uv}$  nicht gewertet.

$\Psi_{uv}$  ist eine Straffunktion, welche die Verschiedenheit in charakt. Eigenschaften der Texturen um  $u$  und  $v$  bewertet. Um eine solche Straffunktion  $\Psi_{uv}$  zu bestimmen, müssen zunächst einmal Abbildungen für die charakteristischen Eigenschaften bestimmt werden. Danach muss noch ein Maß bestimmt werden, mit Hilfe dessen die Unterschiedlichkeit in den charakteristischen Eigenschaften bestimmt werden kann.

*D. Geman* benutzt als charakteristische Eigenschaften von Texturen folgende transformierte Daten:

$\phi^{(1)}(y_s) = y_s$	(Identität)
$\phi^{(2)}(y_s) = \max\{y_t : t \in D\} - \min\{y_t : t \in D\}$	(Intensitäts-Spanne)
$\phi^{(3)}(y_s) =  y_s - \frac{1}{ \partial D } \sum_{t \in \partial D} y_t $	(Residuum)
$\phi^{(4)}(y_s) =  y_s - (y_{s+(1,0)} - y_{s-(1,0)})/2 $	(Richtungsresiduum)
$\phi^{(5)}(y_s) =  y_s - (y_{s+(0,1)} - y_{s-(0,1)})/2 $	(Richtungsresiduum)

Zur Messung der Verschiedenheit in den charakt. Eigenschaften wird der *Kolmogorov-Smirnov-Abstand* benutzt. Man betrachtet die empirische Verteilungsfunktion:

$$F_{\phi^{(i)}(y_D)} : R \mapsto [0, 1], \quad F_{\phi^{(i)}(y_D)}(\tau) = \frac{|\{r \in D : \phi^{(i)}(y_r) \leq \tau\}|}{|D|} \quad (9)$$

Für den *Kolmogorov-Smirnov-Abstand* gilt:

$$d(\phi^{(i)}(y_B), \phi^{(i)}(y_D)) = \max\{|F_{\phi^{(i)}(y_D)}(\tau) - F_{\phi^{(i)}(y_B)}(\tau)|\} \quad (10)$$

Falls der *Kolmogorov-Smirnov-Abstand* der transformierten Daten  $\phi^{(i)}(y_{D_u})$  und  $\phi^{(i)}(y_{D_v})$  für ein  $i$  einen Grenzwert  $c^{(i)}$  überschreitet  
 $\Rightarrow$  Die Texturen in den Regionen  $D_u$  und  $D_v$  werden als unterschiedlich bewertet.

Eine Mögliche Wahl der Straffunktion  $\Psi$ :

$$\Psi_{uv}(y) = \begin{cases} 1 & \text{falls } d(\phi^{(i)}(y_{D_u}), \phi^{(i)}(y_{D_v})) > c^{(i)} \text{ für ein } i \\ -1 & \text{falls } d(\phi^{(i)}(y_{D_u}), \phi^{(i)}(y_{D_v})) \leq c^{(i)} \text{ für alle } i \end{cases} \quad (11)$$

### Label-Label-Interaktion:

Nun soll die Label-Label-Interaktion noch genauer betrachtet werden. Um das Modell zu komplettieren, werden noch Straffunktionen für unerwünschte Labelkonfigurationen definiert.

Unerwünschte Labelkonfigurationen können sein:

Eine Region ist „schwach in  $u$ “.

Eine Region ist „dünn in  $u$ “.

„*schwach in  $u$* “ bedeutet, dass das Label  $l_u$  in den Regionen, die in einer gewissen Umgebung der Region  $U$  liegen, „selten“ vorkommt. Die genaue Ausprägung einer solchen Straffunktion hängt immer vom betrachteten Bild und der Aufgabenstellung ab. Eine mögliche Straffunktion könnte sein:

$$\delta_u^s(l) = \begin{cases} 1 & \text{falls } |\{v \in E_u : l_u = l_v\}| < 9 \\ 0 & \text{sonst} \end{cases} \quad (12)$$

wobei  $E_u$  z.B. ein 5x5 Block in  $U_\varrho$  ist mit Zentrum  $u$

„*dünn in  $u$* “ bedeutet, dass das Label  $l_u$  in vertikaler und/oder horizontaler Richtung isoliert ist. Eine mögliche Straffunktion könnte sein:

$$\delta_u^{th}(l) = \begin{cases} 2 & \text{falls } l_{u-(\varrho,0)} \neq l_u, l_{u+(\varrho,0)} \neq l_u, l_{u-(0,\varrho)} \neq l_u, l_{u+(0,\varrho)} \neq l_u \\ 1 & \text{falls } l_{u-(\varrho,0)} \neq l_u, l_{u+(\varrho,0)} \neq l_u \text{ oder } l_{u-(0,\varrho)} \neq l_u, l_{u+(0,\varrho)} \neq l_u \\ 0 & \text{sonst} \end{cases} \quad (13)$$

Als Gesamtenergie für die Label-Label-Interaktion ergibt sich:

$$V_L(l) = \sum_{u \in U_\varrho} (\delta_u^s(l) + \delta_u^{th}(l)) \quad (14)$$

Die komplette Energiefunktion ergibt sich daraus zu:

$$H(y, l) = K_{PL}(y, l) + V_L(l) \quad (15)$$

$$H(y, l) = \sum_{u \stackrel{\varrho}{\sim} v} \Psi_{uv}(y) \Phi_{uv}(l) + \sum_{u \in U_\varrho} (\delta_u^s(l) + \delta_u^{th}(l)) \quad (16)$$

Durch Minimieren dieser Energiefunktion bekommt man ein optimales Labeling, bzw. eine optimale Segmentierung.



## 4 Textur-Klassifizierung

### 4.1 Grundlagen

Bilder lassen sich in Regionen unterteilen. Diese Regionen enthalten verschiedene Texturen. Die Textur-Klassifizierung dient dazu, die verschiedenen Texturen zu erkennen und die Regionen dementsprechend mit Labeln auszuzeichnen.

Im Gegensatz zur Segmentierung werden also verschiedene Texturen nicht nur getrennt, sondern auch identifiziert. Um einen Ansatz zur Texturklassifizierung nun zu formalisieren, sei  $(y_s)_{s \in S}$  eine Intensitätskonfiguration auf einem endlichem Gitter  $S$ .  $U$  sei ein Teil-Gitter von  $S$  (siehe Abschnitt 3) und  $(D_u)_{u \in U}$  seien Pixelblöcke die  $S$  bedecken.

Nun sei eine Liste von speziellen Texturen gegeben und jede dieser Texturen werde durch ein Label  $l \in L$  repräsentiert. Ziel ist es, passende Labels  $l_D$  zu jedem Block  $D$  zu finden, in dem Sinne, dass das Label, welches einer Region zugeteilt wird, diejenige Textur repräsentiert, die in der Region enthalten ist.

Zunächst soll ein statischer Ansatz zur Texturklassifizierung vorgestellt werden: Der **Minimale-Abstand-Klassifizierer**:

Für jede Referenz-Textur  $j$  mit  $l_j \in L$  repräsentiert der Eigenschaften-Vektor einen Punkt  $P_j \in R^d$ .

$R^d$  wird folgendermaßen in Regionen  $R_j$  aufgeteilt:

$$R_j = \{v \in R^d \mid d(v, P_j) \leq d(v, P_k), \forall k \neq j\} \quad (17)$$

wobei  $d$  der euklidischer Abstand sei.

Ebenso definieren die Eigenschaften eines Pixelblocks einen Punkt  $P_D \in R^d$ .

Falls nun  $P_D \in R_j \Rightarrow$  Die Region  $D$  hat die Textur  $j$ .

Im Folgenden soll nun ein **Bayesscher Ansatz zur Textur-Klassifizierung** vorgestellt werden:

Auch das Labelling selbst stellt ein Muster dar. Daher können Anforderungen bezüglich dessen Struktur in das Modell integriert werden. Daraus ergibt sich, dass man die Bayessche Textur-Klassifizierung in zwei Ansätze unterscheiden kann:

In einen **kontextuellen** und einen **nicht-kontextuellen Ansatz**

**Kontextuell:** Erwartungen an das Labelmuster werden durch eine angemessene apriori-Verteilung mit in das Modell eingebracht.

**Nicht-kontextuell:** Nur die Intensitätskonfiguration der Pixel und eine apriori-Verteilung bzgl. der relativen Häufigkeit der Labels werden verwendet.

## 4.2 Kontextuelle Bayessche Textur-Klassifizierung

Ziel ist es eine Energiefunktion in folgender bekannter Form aufzustellen:

$$H(y, l) = K_{PL}(y, l) + K_L(l) \quad (18)$$

wobei  $K_{PL}(y, l)$  die Energiefunktion der Pixel-Label-Interaktion und  $K_L(l)$  die a priori-Energie der Label-Label-Interaktion darstellen.

### Pixel-Label-Interaktion:

Um die Energiefunktion der Pixel-Label-Interaktion zu bestimmen, verwendet man z.B. die schon bekannte Darstellung der Energie aus dem  $\psi$ -Modell:

$$K_{PL}(y, j) = \sum_{i=1}^d \vartheta_i^{(j)} \sum_{t \sim_s^i} \psi(y_s - y_t), \quad j \in L \quad (19)$$

$(l_u)_{u \in U}$  sei das Labelmuster, wobei wir zur Vereinfachung voraussetzen, dass  $U = S$ .

Unter der Bedingung, dass das gesamte Bild aus reiner Textur  $j$  besteht, erhält man folgendes Texturmodell:

$$\Pi(y, l \mid l_u = j, u \in U) \propto \exp\{K(y, j)\} \quad (20)$$

Die Verteilung der Pixelintensitäten in den Regionen ist lokal unabhängig. Die a-priori-Energie  $K_{PL}$  der Pixel-Label-Interaktion ist demnach auch lokal unabhängig und lässt sich daher als Summe aus lokalen Termen, d.h. blockbezogenen Energien zusammensetzen. Es wird dabei wie folgt vorgegangen: Aus Kenntnis der Energie eines Bildes mit reiner Textur  $j$  (siehe (20)), werden zunächst die Pixelbezogenen Energien abgeleitet:

$$J_{PL}(y, j, s) = \sum_{i=1}^d \vartheta_i^{(j)} \sum_{t \sim_s^i} \psi(y_s - y_t), \quad j \in L \quad (21)$$

Aufbauend darauf wird das Aussehen der blockbezogenen Energien bestimmt:

$$K_{PL}(y, j, u) = \alpha^{-1} \sum_{v \in D_u} J(y, j, v), \quad j \in L \quad (22)$$

wobei  $\alpha$  ein Normierungsfaktor ist, so dass jede Paar-Clique nur einmal auftaucht  $\Rightarrow$  Gesamt-Pixel-Label-Energie für ein Bild mit reiner Textur  $j$  ergibt sich wieder als:

$$K_{PL}(y, j) = \sum_u K(y, j, u) \quad (23)$$

Die Energie der Pixel-Label-Interaktion ergibt sich somit als Summe der blockbezogenen Energien, je nach deren textueller Ausprägung.

$$K_{PL}(y, l) = \sum_u K(y, l_u, u) \quad (24)$$

**Label-Label-Interaktion:**

Durch den Energieterm  $K_L$  werden Erwartungen bezüglich der Struktur des Labellings modelliert.

Wenn man z.B. erwartet das Texturen gebietsweise auftreten, kann die Label-Label-Interaktion wie folgt gewählt werden:

$$K_L(l) = -\eta \sum_{u \sim v} \delta(l_u, l_v), \quad \eta > 0 \quad (25)$$

Für die Gesamtenergie gilt somit:

$$H(y, l) = K_{PL}(y, l) + K_L(l) \quad (26)$$

bzw.:

$$H(y, l) = \sum_u K(y, l_u, u) - \eta \sum_{u \sim v} \delta(l_u, l_v) \quad (27)$$

Der Bayes-Schätzer ergibt sich daher bekanntermaßen als MAP-Schätzer.

**4.3 Nicht-kontextuelle Bayessche Textur-Klassifizierung**

Beim nicht-kontextuellen Ansatz wird keine spezielle Struktur des Labelmusters erwartet. Die a-priori-Verteilung enthält nur Informationen über die relative Häufigkeit der Label. Da die Energien in verschiedene Regionen des Bildes unkorreliert sind, genügt es die a posteriori-Randverteilungen

$$l_s \mapsto \Pi(l_s | y) \quad (28)$$

zu betrachten, und daraus den Bayes-Schätzer als MPM-Schätzer zu bestimmen. Die a-posteriori-Randverteilung hat folgende Form:

$$\Pi(l_s | y) = Z(y)^{-1} \pi(l_s) P_s(y | l_s) \quad (29)$$

wobei  $\Pi(l) = \prod_{s \in U} \pi(l_s)$ , die apriori-Verteilung der Label und  $P_s(y | l_s)$  die bedingte Verteilung von  $l_s$  zu  $y$ .

Durch Maximieren der einzelnen Randverteilungen ergibt sich der Bayes-Schätzer als MPM-Schätzer.

**Anmerkungen:**

Es können noch zusätzlich Labels eingeführt werden:

*doubt-label*: Für Bereiche, in denen Zweifel bezüglich des Labellings herrschen, d.h. in denen die Wahrscheinlichkeit für die Label unter eine gewisse Grenze fallen.

*out-label*: Da die Anzahl verschiedener Texturen in der Regel sehr groß und ungewiss ist oder man nur spezielle Texturen auszeichnen will, weist man allen anderen Bereichen das out-Label zu.

## 5 Anhang

### 5.1 Literaturangaben

G. Winkler. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*. Springer, 2nd ed. 2003

B. Chalmond. *Modelling and Inverse Problems in Image Analysis*. Springer, 2003

<http://de.wikipedia.org>